LEWIS BULLOCK

5A75 2195

DERWENT-ACC-NO: 2001-334062

DERWENT-WEEK: 200135

TITLE: Task scheduling procedure for computer, involves controlling execution order of tasks based on numbers assigned to the tasks

PATENT-ASSIGNEE: MATSUSHITA DENKI SANGYO KK[MATU]

PRIORITY-DATA: 1999JP-0277333 (September 29, 1999)

PATENT-FAMILY:

PUB-NO	PUB-DATE	LANGUAGE	PAGES	MAIN-IPC
JP 2001100807 A	April 13, 2001	N/A	033	G05B 019/00

APPLICATION-DATA:

PUB-NO	APPL-DESCRIPTOR	APPL-NO	APPL-DATE
JP2001100807A	N/A	1999JP-0277333	September 29, 1999

INT-CL (IPC): G05B019/00

ABSTRACTED-PUB-NO: JP2001100807A

BASIC-ABSTRACT:

NOVELTY - Executable tasks are registered in a task execution table. The execution order of the tasks is controlled, based on the number assigned to the tasks.

DETAILED DESCRIPTION - INDEPENDENT CLAIMS are also included for the following:

(a) Task scheduling apparatus;

(b) Program generation and execution procedure;

(c) Program generation and execution apparatus;

(d) Sequence control procedure;

(e) Sequence control apparatus;

(f) Recording medium

USE - For computer.

ADVANTAGE - Modification of execution sequence of tasks can be done in easy
manner.

DESCRIPTION OF DRAWING(S) - The figure shows the block diagram of task
scheduling apparatus. (Drawing includes non-English language text).

CHOSEN-DRAWING: Dwg.1/24

TITLE-TERMS: TASK SCHEDULE PROCEDURE COMPUTER CONTROL EXECUTE ORDER TASK BASED
NUMBER ASSIGN TASK

DERWENT-CLASS: T01 T06

EPI-CODES: T01-F02C; T01-S03; T06-A04;

* NOTICES *

1.This document has been translated by computer. So the translation may not reflect the original precisely.
2.**** shows the word which can not be translated.
3.In the drawings, any words are not translated.

CLAIMS

[Claim(s)]
[Claim 1] The task-scheduling approach characterized by what it has the task activation table on which the task which can be performed is registered, and is performed in order of the task number to which the above-mentioned task which can be performed was given by the task in the task-scheduling approach which controls the execution sequence of the above-mentioned task which can be performed.
[Claim 2] Give a task number to a task and it has the task activation table on which the task which can be performed is registered. It is the task-scheduling approach which controls the execution sequence of the above-mentioned task which can be performed. When the task of the task number corresponding to a current index which can be performed is registered into the above-mentioned task activation table, while performing this task The task-scheduling approach characterized by what it has the 1st step deleted from the above-mentioned task activation table, and the 2nd step which shifts the above-mentioned index, and the above 1st and the 2nd step are repeated for.
[Claim 3] The task-scheduling approach characterized by what the number of tasks performed by one processing is restricted in the task-scheduling approach according to claim 1 or 2, and the above-mentioned processing is periodically performed for.
[Claim 4] The task-scheduling approach characterized by what the number of tasks performed by one processing is restricted for for every group of a task in the task-scheduling approach according to claim 3.
[Claim 5] The task-scheduling approach characterized by what the number of tasks performed by one processing is restricted for for every drive classification of a task in the task-scheduling approach according to claim 3.
[Claim 6] It is the task-scheduling approach characterized by what processing of the time is ended for in the task-scheduling approach of any one publication of claim 3 thru/or claim 5 when the task of the total task number of a task activation table which can be performed is checked by one above-mentioned processing.
[Claim 7] It is the task-scheduling approach characterized by what registration of the task to the above-mentioned task activation table which can be performed is performed for to generating of a task activate request in the task-scheduling approach of any one publication of claim 1 thru/or claim 6.
[Claim 8] The task activation table on which it is task-scheduling equipment which gives a task number to a task and controls the execution sequence of a task using this task number and an index, and the task which can be performed is registered for every task number, The task registration section which registers into the above-mentioned task activation table the task which the task activate request generated as a task which can be performed, Task-scheduling equipment characterized by having the task activation section which performs the task of the task number corresponding to an index which can be performed, shifting an index.
[Claim 9] The program generation method characterized by what the data with which two or more treatment processes contained in a sequence and the transition place from this treatment process were described are read, the contents of the above-mentioned data are changed into a program source code, the above-mentioned program source code is compiled and linked, and an executable program is generated for.
[Claim 10] The program generation method further characterized by what the relation between a task name and the above-mentioned sequence is described for in a program generation method according to claim 9 at the above-mentioned data.
[Claim 11] The program generation method further characterized by what the relation between a task name and an event, the relation between this event and the above-mentioned sequence, and ** are described for in a program generation method according to claim 9 at the above-mentioned data.
[Claim 12] It is the program generation method characterized by what the above-mentioned data are defined for

by the table of a layered structure in the program generation method of any one publication of claim 9 thru/or claim 11.

[Claim 13] Program generation equipment characterized by what it had for the data division two or more treatment processes contained in a sequence and the transition place from this treatment process were described to be, the program generation section which generates a program with reference to the above-mentioned data division, and the write-in section which records the above-mentioned program on a record medium.

[Claim 14] Program generation equipment further characterized by what the relation between a task name and the above-mentioned sequence is described for in program generation equipment according to claim 13 at the above-mentioned data division.

[Claim 15] Program generation equipment further characterized by what the relation between a task name and an event, the relation between this event and the above-mentioned sequence, and ** are described for in program generation equipment according to claim 13 at the above-mentioned data division.

[Claim 16] It is program generation equipment characterized by what the above-mentioned data division are defined for by the table of a layered structure in the program generation equipment of any one publication of claim 13 thru/or claim 15.

[Claim 17] The program execution approach characterized by what it has the sequence information two or more treatment processes contained in a sequence and the transition place from this treatment process were described to be, and it is the program execution approach of performing sequence control, sequential execution of the process of sequence control is carried out with reference to the above-mentioned sequence information, and degree treatment process is performed for with reference to the above-mentioned sequence information based on the activation result of the above-mentioned process.

[Claim 18] The sequence information the transition place from two or more treatment processes and these treatment processes which are contained in a sequence was described to be, It is the program execution approach which carries out sequential execution of the process which has the activation table which registers the process which can be performed and was registered into the above-mentioned activation table, and which can be performed. While performing the 1st step which registers the head process of sequence control into the above-mentioned activation table with reference to the above-mentioned sequence information as a process which can be performed, and the above-mentioned process which can be performed It is based on the 2nd step deleted from the above-mentioned activation table, and the activation result of the above-mentioned process. The program execution approach characterized by what the above 2nd and the 3rd step are repeated for until it has the 3rd step which registers degree treatment process of the above-mentioned sequence information into an activation table as a process which can be performed and degree treatment process of the above-mentioned sequence information is lost.

[Claim 19] It is the program execution approach characterized by what the above-mentioned sequence information has the latency time corresponding to a process in the program execution approach according to claim 18, it is the 2nd step of the above and only the above-mentioned latency time does for the waiting for time amount before activation of the above-mentioned process.

[Claim 20] The sequence information with which gave the task number to the task and the transition place from the relation between each task name and a sequence, two or more treatment processes which can be set to this sequence, and this treatment process was described to be, The activation table on which the process which can be performed is registered for every task to which this process belongs, It is the program execution approach which carries out sequential execution of the process which ****(ed) and was registered into the above-mentioned table which can be performed, and which can be performed while carrying out scheduling of the task. When performing one of sequence control The process included in this sequence control is registered into an activation table as a process which can be performed. The above-mentioned process which can be performed is performed in order of a task number, and degree treatment process is registered into the above-mentioned activation table with reference to the above-mentioned sequence information as a process which can be performed based on the activation result of the above-mentioned process. Registration of the above-mentioned process which can be performed, The program execution approach characterized by what activation of this process is repeated and is performed for.

[Claim 21] The sequence information with which gave the task number to the task and the transition place from the relation between each task name and a sequence, the relation between the above-mentioned sequence and an event, two or more treatment processes contained in the above-mentioned sequence, and this treatment process was described to be, The activation table on which the process which can be performed is registered for every

task to which this process belongs, It has an event queue for every task number. A task A task number, It is the program execution approach which carries out sequential execution of the process which was registered into the above-mentioned activation table while carrying out scheduling using the index, and which can be performed. When a registration event is in the event queue of the same task number as a current index and there is no process of this task number which can be performed in the above-mentioned activation table The 1st step which deletes the above-mentioned registration event and registers the head process of the sequence corresponding to this event into the above-mentioned activation table with reference to the above-mentioned sequence information as a process of the above-mentioned task number which can be performed, When the process of the same task number as a current index which can be performed is registered into the above-mentioned activation table The 2nd step deleted from the above-mentioned activation table while performing this process, When there is degree treatment process with reference to the above-mentioned sequence information based on the activation result of the above-mentioned process The program execution approach characterized by having the 3rd step which registers this following treatment process into the above-mentioned activation table as a process which can be performed, and the 4th step which shifts an index, and repeating the above 1st thru/or the 4th step.

[Claim 22] In the program execution approach according to claim 21 at the 1st step of the above When a registration event is in the event queue of the same task number as a current index and the process of this task number which can be performed is shown in the above-mentioned activation table The program execution approach characterized by what can be set up by transition authorization of the above-mentioned process which can be performed for whether it replaces with this process that can be performed and the head process of the sequence corresponding to the head event of the above-mentioned event queue is registered into the above-mentioned activation table.

[Claim 23] It is the program execution approach characterized by to have the step which also registers the latency time corresponding to this process in the program execution approach according to claim 21 or 22 when registering into the above-mentioned activation table the process which can be performed, does not perform this process at the time of activation of this process when the above-mentioned latency time is not 0, but updates the latency time of the above-mentioned activation table after the 4th step of the above.

[Claim 24] The program execution approach characterized by what the number of processes performed by one processing is restricted in the program execution approach of any one publication of claim 20 thru/or claim 23, and the above-mentioned processing is periodically performed for.

[Claim 25] The program execution approach characterized by what the number of processes performed by one processing is restricted for for every group of a task in the program execution approach according to claim 24.

[Claim 26] The program execution approach characterized by what the number of processes performed by one processing is restricted for for every drive classification of a task in the program execution approach according to claim 24.

[Claim 27] It is the program execution approach characterized by what processing of the time is ended for in the above-mentioned processing in the program execution approach of any one publication of claim 24 thru/or claim 26 when the process of the total task number of the above-mentioned activation table which can be performed is checked.

[Claim 28] It is the program execution approach characterized by what the above-mentioned sequence information is an executable program in the program execution approach of any one publication of claim 17 thru/or claim 27.

[Claim 29] It is the program execution approach characterized by what is been the executable program by which the above-mentioned sequence information was recorded on the record medium in the program execution approach of any one publication of claim 17 thru/or claim 28.

[Claim 30] It is the program execution approach characterized by what the above-mentioned sequence information is defined for by the table of a layered structure in the program execution approach of any one publication of claim 17 thru/or claim 29.

[Claim 31] Program execution equipment characterized by what it had the program execution section which performs sequence control for by the program read the account of a top with the reading section which reads the program written in the record medium.

[Claim 32] It is program execution equipment characterized by what the sequence information two or more treatment processes by which the above-mentioned program is included in a sequence in program execution equipment according to claim 31, and the transition place from this treatment process were described to be is

included for.

[Claim 33] In program execution equipment according to claim 32 the above-mentioned program execution section The activation table on which the process which can be performed is registered, and the process registration section which registers into the above-mentioned activation table the process which can be performed with reference to the above-mentioned program, Program execution equipment characterized by having the processes run section which the above-mentioned process which can be performed is performed [section], and it judges [section] whether the activation result is normal, and makes a process [degree] process or a transition place process register into the above-mentioned process registration section as a process which can be performed.

[Claim 34] It is program execution equipment characterized by including the sequence information two or more treatment processes by which the above-mentioned program is included in the relation between each task name and a sequence, the relation between the above-mentioned sequence and an event, and the above-mentioned sequence in program execution equipment according to claim 31, and the transition place from this treatment process were described to be.

[Claim 35] It is program execution equipment characterized by the above-mentioned program execution section performing scheduling of a task with activation of sequence control in program execution equipment according to claim 34 using a task number and an index.

[Claim 36] In program execution equipment according to claim 34 or 35 the above-mentioned program execution section The activation table on which the task number was given for the process which can be performed and which is registered for every task, The process registration section which registers into the above-mentioned activation table the process which can be performed with reference to the above-mentioned program, The event queue table on which the generated event has the event queue with which this event belongs, and which is registered for every task, The event analysis section which will register this event into the event queue corresponding to the task to which this event belongs with reference to the above-mentioned program if an event occurs, An instruction is given to the process registration section so that it may register with the above-mentioned activation table by making each process of the sequence control corresponding to a registration event into the process which can be performed. Program execution equipment characterized by having the sequence activation section which performs the process of the same task number as this index which can be performed while shifting an index.

[Claim 37] It is program execution equipment characterized by the above-mentioned program making the layered structure in the program execution equipment of any one publication of claim 31 thru/or claim 36.

[Claim 38] It has data with which two or more treatment processes contained in a sequence and the transition place from this treatment process were described. Are the sequence control approach of performing the above-mentioned sequence control, and the contents of the above-mentioned data are read. The sequence control approach which changes into a program, stores in the storage section, carries out sequential execution of each process of sequence control with reference to the program of the above-mentioned storage section, and is characterized by what degree treatment process is performed for with reference to the above-mentioned program based on the activation result of the above-mentioned process.

[Claim 39] The data with which the task number was given to the task and the transition place from the relation between each task name and a sequence, the relation between the above-mentioned sequence and an event, two or more treatment processes contained in the above-mentioned sequence, and this treatment process was described, The activation table on which the process which can be performed is registered for every task to which this process belongs, It has an event queue for every task number. A task A task number, The 1st step which is the sequence control approach which carries out sequential execution of the above-mentioned activation ****, carrying out scheduling using an index, reads the contents of the above-mentioned data, changes into a program, and is stored in the storage section, When a registration event is in the event queue of the same task number as a current index and there is no process of this task number which can be performed in the above-mentioned activation table The 2nd step which deletes the above-mentioned registration event and registers the head process of the sequence corresponding to this event into the above-mentioned activation table with reference to the above-mentioned sequence information as a process of the above-mentioned task number which can be performed, When the process of the same task number as a current index which can be performed is registered into the above-mentioned activation table The 3rd step deleted from the above-mentioned activation table while performing this process, When there is degree treatment process with reference to the above-mentioned sequence information based on the activation result of the above-mentioned process The

sequence control approach characterized by what it has the 4th step which registers this following treatment process into the above-mentioned activation table as a process which can be performed, and the 5th step which shifts an index, and the above 2nd thru/or the 5th step are repeated for.

[Claim 40] It is the sequence control approach characterized by what the above-mentioned data are defined for by the table of a layered structure in the sequence control approach according to claim 38 or 39.

[Claim 41] The data division two or more treatment processes contained in a sequence and the transition place from this treatment process were described to be, The program generation section which reads the contents of the above-mentioned data division and is changed into a program, With reference to the above-mentioned program stored in the storage section in which the above-mentioned program is stored, and the above-mentioned storage section, sequential execution of each process of sequence control is carried out. Sequence control equipment characterized by having the program execution section which performs degree treatment process with reference to the above-mentioned program based on the activation result of the above-mentioned process.

[Claim 42] In sequence control equipment according to claim 41 the above-mentioned program execution section The activation table on which the process which can be performed is registered, and the process registration section which registers into the above-mentioned activation table the process which can be performed with reference to the program of the above-mentioned storage section, Sequence control equipment characterized by having the processes run section which the above-mentioned process which can be performed is performed [section], and the activation result judges [section] whether it is normal, and makes a process [degree] process or a transition place process register into the above-mentioned process registration section as a process which can be performed.

[Claim 43] The relation between each task name and a sequence, and the relation between the above-mentioned sequence and an event, The data division two or more treatment processes contained in the above-mentioned sequence and the transition place from this treatment process were described to be, The contents of the above-mentioned data division are read and the above-mentioned program stored in the program generation section changed into a program, the storage section in which the above-mentioned program is stored, and the above-mentioned storage section is referred to. With activation of sequence control Sequence control equipment characterized by having the program execution section which also performs scheduling of a task using a task number and an index.

[Claim 44] In sequence control equipment according to claim 43 the above-mentioned program execution section The activation table on which the task number was given for the process which can be performed and which is registered for every task, The process registration section which registers into the above-mentioned activation table the process which can be performed with reference to the above-mentioned program, The event queue table on which the generated event has the event queue with which this event belongs, and which is registered for every task, The event analysis section which will register this event into the event queue corresponding to the task to which this event belongs with reference to the above-mentioned program if an event occurs, An instruction is given to the process registration section so that it may register with the above-mentioned activation table by making each process of the sequence control corresponding to a registration event into the process which can be performed. Sequence control equipment characterized by having the sequence activation section which performs the process of the same task number as this index which can be performed while shifting an index.

[Claim 45] It is sequence control equipment characterized by the above-mentioned data division being defined by the table of a layered structure in the sequence control equipment of any one publication of claim 41 thru/or claim 44.

[Claim 46] The task-scheduling program documentation medium characterized by what it has the task activation table on which the task which can be performed is registered, and the program performed in order of the task number to which the above-mentioned task which can be performed was given by the task in the task-scheduling program documentation medium which recorded the program which controls the execution sequence of the above-mentioned task which can be performed was recorded for.

[Claim 47] The program generator record medium characterized by recording the program which reads the data with which two or more treatment processes contained in a sequence and the transition place from this treatment process were described, the program which changes the contents of the above-mentioned data into a program source code, and the program which compiles and links the above-mentioned program source code, and generates an executable program.

[Claim 48] It has the sequence information two or more treatment processes contained in a sequence and the transition place from this treatment process were described to be. The program which is the program execution program documentation medium which recorded the program which performs sequence control, and carries out sequential execution of the process of sequence control with reference to the above-mentioned sequence information, The program execution program documentation medium characterized by recording the program which performs degree treatment process with reference to the above-mentioned sequence information based on the activation result of the above-mentioned process.

[Claim 49] The sequence information with which gave the task number to the task and the transition place from the relation between each task name and a sequence, the relation between the above-mentioned sequence and an event, two or more treatment processes contained in the above-mentioned sequence, and this treatment process was described to be, The activation table on which the process which can be performed is registered for every task to which this process belongs, It has an event queue for every task number. A task A task number, It is the program execution program documentation medium which recorded the program which carries out sequential execution of the above-mentioned process which can be performed while carrying out scheduling using an index. When a registration event is in the event queue of the same task number as a current index and there is no process of this task number which can be performed in the above-mentioned activation table The 1st step which deletes the above-mentioned registration event and registers the head process of the sequence corresponding to this event into the above-mentioned activation table with reference to the above-mentioned sequence information as a process of the above-mentioned task number which can be performed, When the process of the same task number as a current index which can be performed is registered into the above-mentioned activation table The 2nd step deleted from the above-mentioned activation table while performing this process, When there is degree treatment process with reference to the above-mentioned sequence information based on the activation result of the above-mentioned process The program execution program documentation medium characterized by what it had the 3rd step which registers this following treatment process into the above-mentioned activation table as a process which can be performed, and the 4th step which shifts an index, and the program which repeats the above 1st thru/or the 4th step was recorded for.

[Claim 50] It has data with which two or more treatment processes contained in a sequence and the transition place from this treatment process were described. The program which is the sequence control program documentation medium which recorded the program which performs the above-mentioned sequence control, reads the contents of the above-mentioned data, changes into a program, and is stored in the storage section, The program which carries out sequential execution of each process of sequence control with reference to the program of the above-mentioned storage section, The sequence control program documentation medium characterized by recording the program which performs degree treatment process with reference to the above-mentioned program based on the activation result of the above-mentioned process.

[Claim 51] The record medium which recorded the sequence-control data which are the record medium which recorded as data two or more treatment processes contained in a sequence, and the transition place from this treatment process, and are characterized by what it had for the 1st field which comes to record two or more above-mentioned treatment processes, and the 2nd field which relates with each of two or more above-mentioned treatment processes, and comes to record an unit or two or more transition place processes.

[Claim 52] It is the record medium which recorded the sequence control data characterized by what it had for the 3rd field where the above-mentioned record medium comes to record the latency time of the above-mentioned process in relation to the above-mentioned process further in a record medium according to claim 51.

[Claim 53] A task name, the event corresponding to this task name, and the sequence corresponding to this event, Two or more treatment processes which can be set to this sequence, and the transition place from this treatment process, The 1st field which is the record medium recorded as data and comes to record the above-mentioned task name, The 2nd field which relates with the above-mentioned task name and comes to record an unit or two or more events, It relates with each of the above-mentioned event, and has the 3rd field which comes to record a sequence. The 3rd field of the above Furthermore, the process record section which comes to record the process which relates with each of the above-mentioned sequence and is included in the above-mentioned sequence, the transition place process record section which the above-mentioned process is alike, respectively, and relates and comes to record an unit or two or more transition place processes -- since -- the record medium which recorded the sequence control data characterized by what is constituted.

[Translation done.]

* NOTICES *

1.This document has been translated by computer. So the translation may not reflect the original precisely.
2.**** shows the word which can not be translated.
3.In the drawings, any words are not translated.

DETAILED DESCRIPTION

[Detailed Description of the Invention]
[0001]
[Field of the Invention] This invention relates to the task-scheduling approach which does not need fine control especially, task-scheduling equipment, a program generation method, program generation equipment, the program execution approach, program execution equipment, the sequence control approach, sequence control equipment, and a record medium.
[0002]
[Description of the Prior Art] In a microcomputer control program, the sequence control which carries out sequential execution of the operations sequence (sequence) decided beforehand is common, carrying out coordination actuation of the hardware of the connected circumference. In this sequence control, the state transition of being as redoing a certain processing **** [, and] occurs. [interrupting processing, if an error occurs for every processing] A sequence is assigned to the task which is the run unit of CPU, and it is made to operate in the program of a microcomputer, carrying out scheduling of two or more tasks. If a task is classified according to drive classification, it will become three kinds such as the resident task which is always operating, the fixed-cycle task started for every fixed time amount the period beforehand decided using timer ability etc., and the event task started by external events, such as a user input, from the time of a system startup.
[0003] First, the conventional task-scheduling approach is explained. As the conventional task-scheduling approach, round robin scheduling is common. This round robin scheduling is a method which assigns CPU equally to them, when it performs from what has an execution priority high when there are two or more tasks from which an execution priority differs and there are two or more tasks with an equal execution priority. that is, if CPU is assigned to a certain task and only the decided fixed time quantum passes when there are two or more tasks with an equal execution priority, the task will be terminated compulsorily -- having -- the following task with an equal execution priority -- receiving -- the above -- only a fixed time quantum is the method that CPU is assigned.
[0004] For example, CPU which is not illustrated performs processing every 5ms, and the relation between a task and its execution priority presupposes that it is given in the table of drawing 20 (a). When a task activate request occurs, each task is registered into a task activation table, and the registered task is performed by order with a high execution priority, i.e., descending of a value. Before CPU starts the 1st processing at 0ms, suppose that Task A - Task E were registered into the task activation table like drawing 20 (b) in this sequence. In this registration, a task is registered into a task activation table from top one as a task which can be performed. And a task is performed from a top thing and the performed task is deleted from a task activation table. When a top task is performed and deleted, the remaining tasks will shift to one top one. And if the high task of an execution priority which can be performed is lost, the task of the following execution priority which can be performed is performed. In addition, it is shown in drawing 20 (b) and (c) that "-" does not have the task which is registered and which can be performed. Here, since it is easy, it is [that generating of the task activate request of each task and registration to a task activation table are only performed once, and], and the task activate request of each task already presupposes that it did not generate more than this.
[0005] First, CPU performs the highest task A and Task B of an execution priority in the 1st processing for 0ms, and deletes them from a task activation table. Next, the task C with an execution priority high to the 2nd and Task D are performed, and they are also deleted from a task activation table. Next, the task E with an execution priority low No. 1 is performed, and this task E is deleted from a task activation table. And the 1st processing is ended. Next, since there is no task which was registered into the task activation table like drawing

20 (c) at the time of initiation of the 2nd processing 5ms after and which can be performed, only the check of a task activation table is performed.

[0006] Thus, a task can be performed in order by setting an execution priority as a task. By the task-scheduling approach by above-mentioned round robin scheduling, when there are two or more fixed-cycle tasks started periodically, or when a task is registered into a task activation table beyond the throughput of CPU, a specific task with a low execution priority may not be performed. In order to avoid such a situation, a designer needs to create the program for task scheduling by which each task is performed equally by setting up the starting period and the execution priority or changing the execution priority of a task dynamically for every fixed period so that the starting timing and the execution priority of all tasks may be examined beforehand and there may be no task which is not performed.

[0007] Next, the program generation method of the conventional sequence control is explained. As a program generation method of sequence control, the approach of using a fixed sequence pattern, and the method of using a state transition table are learned. First, the program generation method using a fixed sequence pattern is explained. The program generation method using a fixed sequence pattern is the approach of fixing the whole structure as a sequence pattern, creating a sequence sequence table, an operating-condition table, etc. to the possible part of modification, making combine automatically these each table information and sequence patterns as the whole structure, and generating a sequence control program.

[0008] Hereafter, an example explains the program generation method using a fixed sequence pattern. First, the sequence sequence table of drawing 21 (a) and the operating-condition table of the processing B of drawing 21 R> 1 (b) are defined. Processing 1 - processing 3 are defined as Processing A - Processing C, conditions 1 - conditions 3, and a list, the contents of processing which interpret processing of a publication and the contents of a condition on this table, and correspond based on such table information, conditional branching, etc. include in the sequence pattern which it has beforehand, and the program generation method using a fixed sequence pattern generates a sequence-control program automatically as a whole in these tables.

[0009] Drawing 22 is a flow chart which shows an example of the sequence control program generated based on drawing 21 (a) and each table information on (b). Next, the sequence control program generated by the program generation method which uses a fixed sequence pattern using the flow chart of drawing 22 is explained. First, since the initiation latency time of Processing A is 0ms when processing is started, CPU performs processing A (step S2201). Next, waiting for time amount only of the 20ms only of the initiation latency times over Processing B is carried out (step S2202), they progress to step S2203, and it judges whether an operating condition fulfills conditions 1. Processing 1 will be performed if it is filling (step S2204). While not filling, or when processing 1 is performed, it judges whether an operating condition fulfills conditions 2 (step S2205). Processing 2 will be performed if it is filling (step S2206). While not filling, or when processing 2 is performed, it judges whether an operating condition fulfills conditions 3 (step S2207). Processing 3 will be performed if it is filling (step S2208). While not filling, or when processing 3 is performed, waiting for time amount only of the 5ms only of the initiation latency times over Processing C is carried out (step S2209), then they perform processing C (step S2210), and end all processings.

[0010] In addition, since actuation of equipment etc. is completed, or since equipment etc. is stabilized in the condition that the next processing can be performed, the initiation latency time over each processing in the above-mentioned explanation means fixed time amount defined to each processing, and waiting. Although it is also possible whether processing can be performed and or not for equipment to be supervised continuously, the configuration of equipment of the configuration of a program will also be simple, and the direction which sets up the initiation latency time will end. Thus, according to the program generation method using a fixed sequence pattern, the sequence control program shown by the flow chart of drawing 22 is automatically generable by indicating the desired contents of processing and the desired initiation latency time to drawing 21 (a), and indicating the contents of processing when fulfilling desired conditions and conditions on the operating-condition table of Processing B.

[0011] Next, the program generation method using a state transition table is explained. Drawing 23 is drawing showing the state transition table which expressed sequence control by the condition (S) and the matrix which consists of an event (E). First, the state transition table of drawing 23 defines a control sequence. In this table, the upper case of each cel shows the contents of processing, and below the arrow head of the lower berth shows the transition place of a condition (S). In addition, the contents of processing show that the cel of "-" does not process, and the cel of a null shows performing neither processing of what nor a state transition. It defines in which condition if which event happens in each condition, what kind of processing will be performed by the

above-mentioned state transition table, and after the processing is completed, it changes. The approach of generating automatically those processings and the program which described transition of a condition using the above-mentioned state transition table is a program generation method using a state transition table.

[0012] The above-mentioned program generation method interprets the written contents of the table with reference to the state transition table of drawing 23 . A certain condition (S) when a certain processing or transition exists in a certain event (E) It is the approach of generating a sequence control program by specifying those conditions (S) and events (E) by conditional branching, and a program's describing that the processing indicated and transition are performed, and performing it to the whole state transition table.

[0013] Drawing 24 is a flow chart which shows an example of the sequence control program generated based on the state transition table of drawing 23 . In addition, the contents of the control sequence are the same as explanation of the program generation method using a fixed sequence pattern. The sequence control program generated by the program generation method which uses a state transition table using the flow chart of drawing 24 is explained. First, if a program is started, it will carry out under a condition = halt (step S2401). Next, since it judged in **** during the condition = halt (step S2402) and carried out under the condition = halt at step S2401, it progresses to step S2403 and judges in event = sequence initiation and ****. Event = at the time of sequence initiation and **, processing A is performed and it considers as waiting for condition =20ms (step S2404). It is step S2402, and during a condition = halt, when it is judged that it does not come out, or when it is judged at step S2403 that an event is not sequence initiation, and when processing of step S2404 is performed, it judges in waiting and **** for condition =20ms (step S2405). condition = -- the waiting for 20ms -- it comes out, and if it is, it will judge in progress and **** for event =20ms (step S2406), and will consider as condition = processing A termination at the time of event =20ms progress and ** (step S2407). It is step S2405, and the waiting for condition =20ms, when it is judged that it does not come out, when not being judged as waiting for event =20ms at step S2406, or when step S2407 is performed, it judges in condition = processing A termination and **** (step S2408).

[0014] Condition = at the time of processing A termination and **, it judges in the event = conditions 1 and **** (step S2409). Event = processing 1 is performed at the time of conditions 1 and **, and it makes it waiting for condition =5ms (step S2410). Moreover, the event = conditions 1, when not coming out, or when processing of step S2410 is performed, it judges in the event = conditions 2 and **** (step S2411). Event = at the time of conditions 2 and **, processing 2 is performed and it considers as waiting for condition =5ms (step S2412). Event = conditions 2, when not coming out, or when processing of step S2412 is performed, it judges in the event = conditions 3 and **** (step S2413). Event = processing 3 is performed at the time of conditions 3 and **, and it makes it waiting for condition =5ms (step S2414).

[0015] step S2408 -- condition = -- when [the event = conditions 3 and when / when it processed A ends and it is judged that it does not come out, or / not coming out], or, when processing of step S2414 is performed, it judges in waiting and **** for condition =5ms (step S2415). It is an event when a condition is the waiting for 5ms. It judges in progress and **** for =5ms (step S2416), and considers as condition = processing B termination at the time of event =5ms progress and ** (step S2417). step S2415 -- setting -- the waiting for condition =5ms, and the time of not coming out -- step S2416 -- setting -- event = -- for 5ms, it passes, and when not coming out, or when step S2417 is performed, it judges in condition = processing B termination and **** (step S2418). Condition = it judges in event = processing B termination and **** at the time of processing B termination and ** (step S2419), and processing C is performed at the time of processing B termination, and it considers it as condition = processing C termination (step S2420). step S2418 -- setting -- condition = -- the time of not processing, B ending and coming out -- step S2419 -- setting -- event = -- it processes B ends, and when not coming out, or when step S2420 is performed, it judges in condition = processing C termination and **** (step S2421). Condition = a program is ended at the time of processing C termination and **. however -- otherwise, -- coming -- being alike -- the processing from return and step S2401 is again repeated to step S2401.

[0016] Thus, according to the program generation method using a state transition table, sequence control is expressed by the matrix which consists of the condition (S) and event (E) like drawing 23 . By generating a program like the flow chart of drawing 24 which specifies a condition (S) and an event (E) by conditional branching using the above-mentioned state transition table, and performs processing and transition When an addition, deletion, modification, etc. change an event (E) into a condition (S) in the above-mentioned state transition table, easily, the addition of the processing in sequence control and deletion are performed, and a change etc. can be made, and can offer the high program generation method of versatility and description

capacity. In addition, the sequence control program generated by the above-mentioned program generation method is compiled and linked, and turns into a program which can be performed.

[0017]

[Problem(s) to be Solved by the Invention] As mentioned above, in order that the round robin scheduling which is the conventional task-scheduling approach might carry out scheduling of the task with an execution priority, when many task activate requests occurred at once, only the task with a high execution priority was performed by processing at each time, and the task with a low execution priority had the problem that it did not perform. For example, if the event task of priority is higher than a fixed-cycle task when the activate request of two or more fixed-cycle tasks started periodically and two or more event tasks occurs in coincidence, the case where a fixed-cycle task cannot operate an assignment period will occur. In order to avoid this, a designer needs to constitute a program or needs to change the execution priority of a task dynamically every activation of a task, and a fixed period so that the execution priority, the starting timing, and the starting period of all tasks may be examined beforehand and there may be no task which is not performed, and it is necessary to constitute a program so that a task may be performed equally. However, by these solution approaches, there was a problem that the burden of the designer of a task-scheduling program increased, and a program will become complicated, and the capacity of a program became large as the result.

[0018] Furthermore, by the conventional task-scheduling approach, in order to perform in order the task which was registered into the task activation table and which can be performed, for example, even if it performed processing every 5ms, when required for activation of all tasks 5ms or more, about a resident task or a fixed-cycle task, there was also a problem that the starting timing of the following task was not secured.

[0019] Moreover, the program generation method using a fixed sequence pattern can respond only to the sequence pattern decided beforehand. That is, in explanation in the conventional example, although Processing A can be changed into processing A', or it newly adds Processing D, it cannot change so that Processing C may be divided into conditions 4 - conditions 6 and processing 4 - processing 6 may be performed. Thus, the program generation method using a fixed sequence pattern had the narrow adaptation range, and since it needed to change a sequence pattern about the addition of processing etc., it had the problem that the program generation method itself had to be changed.

[0020] Moreover, although the program generation method using a state transition table had high description capacity as mentioned above and it could respond to the addition of processing, modification, etc. flexibly, the generated program had the problem that became the thing of the redundant expression with much conditional branching, and the capacity of a program became large. Furthermore, since the state transition table created since a condition (S) and one event (E) increase became large and increased conditional branching etc. by the addition of time amount conditions, such as an addition of processing, or the latency time, there was also a problem that the capacity of a program will increase. For example, when flow chart drawing 22 and drawing 24 expressing the same sequence control of a program are compared, it turns out that the program created using the state transition table is more complicated than the program created using the fixed sequence pattern. Moreover, since the activity which creates a state transition table turned into an activity which creates a program with an equal mostly after all when there is much conditional branching etc., there was also a problem that the advantage as a program generation method was lost.

[0021] It aims at offering the task-scheduling approach by which it is made in order that this invention may solve the above-mentioned trouble, and the situation that a specific task is not performed can be canceled, and the starting timing of a resident task and a fixed-cycle task is secured, and equipment. Moreover, this invention aims at offering the task-scheduling program documentation medium which recorded the program corresponding to the above-mentioned task-scheduling approach. Moreover, the capacity of a data program is small and this invention aims at offering the sequence control approach which can respond to the addition of a process, modification, etc. easily, and equipment. Moreover, this invention aims at offering the sequence control program documentation medium which recorded the program corresponding to the above-mentioned sequence control approach. Moreover, this invention aims at offering the program generation method which can generate the program as data with a small capacity, and equipment. Moreover, this invention aims at offering the program generator record medium which recorded the program corresponding to the above-mentioned program generation method. Moreover, this invention aims at offering the program execution approach that a data program can be performed as sequence control, and equipment. Moreover, this invention aims at offering the program execution program documentation medium which recorded the program corresponding to the above-mentioned program execution approach. Moreover, this invention aims at offering the record medium which

recorded the data which have the layered structure used in sequence control.
[0022]
[Means for Solving the Problem] In order to attain the above-mentioned purpose, the task-scheduling approach concerning claim 1 has the task activation table on which the task which can be performed is registered, and performs it in order of the task number to which the above-mentioned task which can be performed was given by the task in the task-scheduling approach which controls the execution sequence of the above-mentioned task which can be performed.
[0023] Moreover, the task-scheduling approach concerning claim 2 Give a task number to a task and it has the task activation table on which the task which can be performed is registered. It is the task-scheduling approach which controls the execution sequence of the above-mentioned task which can be performed. When the task of the task number corresponding to a current index which can be performed is registered into the above-mentioned task activation table, while performing this task It has the 1st step deleted from the above-mentioned task activation table, and the 2nd step which shifts the above-mentioned index, and the above 1st and the 2nd step are repeated.
[0024] Moreover, in the task-scheduling approach according to claim 1 or 2, the task-scheduling approach concerning claim 3 restricts the number of tasks performed by one processing, and performs the above-mentioned processing periodically.
[0025] Moreover, the task-scheduling approach concerning claim 4 restricts the number of tasks performed by one processing for every group of a task in the task-scheduling approach according to claim 3.
[0026] Moreover, the task-scheduling approach concerning claim 5 restricts the number of tasks performed by one processing for every drive classification of a task in the task-scheduling approach according to claim 3.
[0027] Moreover, in the task-scheduling approach of any one publication of claim 3 thru/or claim 5, the task-scheduling approach concerning claim 6 is one above-mentioned processing, and when the task of the total task number of a task activation table which can be performed is checked, it ends processing of the time.
[0028] Moreover, the task-scheduling approach concerning claim 7 performs registration of the task to the above-mentioned task activation table which can be performed to generating of a task activate request in the task-scheduling approach of any one publication of claim 1 thru/or claim 6.
[0029] Moreover, the task-scheduling equipment concerning claim 8 The task activation table on which it is task-scheduling equipment which gives a task number to a task and controls the execution sequence of a task using this task number and an index, and the task which can be performed is registered for every task number, It has the task registration section which registers into the above-mentioned task activation table the task which the task activate request generated as a task which can be performed, and the task activation section which performs the task of the task number corresponding to an index which can be performed while shifting an index.

[0030] Moreover, the program generation method concerning claim 9 reads the data with which two or more treatment processes which can be set to sequence control, and the transition place from this treatment process were described, changes the contents of the above-mentioned data into a program source code, compiles and links the above-mentioned program source code, and generates an executable program.
[0031] Moreover, in the program generation method according to claim 9, as for the program generation method concerning claim 10, the relation between a task name and the above-mentioned sequence is further described by the above-mentioned data.
[0032] Moreover, in the program generation method according to claim 9, as for the program generation method concerning claim 11, the relation between a task name and an event, the relation between this event and the above-mentioned sequence, and ** are further described by the above-mentioned data.
[0033] Moreover, in the program generation method of any one publication of claim 9 thru/or claim 11, the above-mentioned data are defined for the program generation method concerning claim 12 by the table of a layered structure.
[0034] Moreover, the program generation equipment concerning claim 13 is equipped with the data division two or more treatment processes contained in a sequence and the transition place from this treatment process were described to be, the program generation section which generates a program with reference to the above-mentioned data division, and the write-in section which records the above-mentioned program on a record medium.
[0035] Moreover, in program generation equipment according to claim 13, as for the program generation equipment concerning claim 14, the relation between a task name and the above-mentioned sequence is further

described by the above-mentioned data division.
[0036] Moreover, in program generation equipment according to claim 13, as for the program generation equipment concerning claim 15, the relation between a task name and an event, the relation between this event and the above-mentioned sequence, and ** are further described by the above-mentioned data division.
[0037] Moreover, in the program generation equipment of any one publication of claim 13 thru/or claim 15, the above-mentioned data division are defined for the program generation equipment concerning claim 16 by the table of a layered structure.
[0038] Moreover, the program execution approach concerning claim 17 has the sequence information two or more treatment processes contained in a sequence and the transition place from this treatment process were described to be, is the program execution approach of performing sequence control, carries out sequential execution of the process of sequence control with reference to the above-mentioned sequence information, and performs degree treatment process with reference to the above-mentioned sequence information based on the activation result of the above-mentioned process.
[0039] Moreover, the program execution approach concerning claim 18 The sequence information the transition place from two or more treatment processes and these treatment processes which are contained in a sequence was described to be, It is the program execution approach which carries out sequential execution of the process which has the activation table which registers the process which can be performed and was registered into the above-mentioned activation table, and which can be performed. While performing the 1st step which registers the head process of sequence control into the above-mentioned activation table with reference to the above-mentioned sequence information as a process which can be performed, and the above-mentioned process which can be performed It is based on the 2nd step deleted from the above-mentioned activation table, and the activation result of the above-mentioned process. The above 2nd and the 3rd step are repeated until it has the 3rd step which registers degree treatment process of the above-mentioned sequence information into an activation table as a process which can be performed and degree treatment process of the above-mentioned sequence information is lost.
[0040] Moreover, it has the latency time on the program execution approach according to claim 18 and corresponding to [approach / concerning claim 19 / program execution] a process in the above-mentioned sequence information, and it is the 2nd step of the above and only the above-mentioned latency time carries out waiting for time amount before activation of the above-mentioned process.
[0041] Moreover, the program execution approach concerning claim 20 The sequence information with which gave the task number to the task and the transition place from the relation between each task name and a sequence, two or more treatment processes which can be set to this sequence, and this treatment process was described to be, The activation table on which the process which can be performed is registered for every task to which this process belongs, It is the program execution approach which carries out sequential execution of the process which ****(ed) and was registered into the above-mentioned table which can be performed, and which can be performed while carrying out scheduling of the task. When performing one of sequence control The process included in this sequence control is registered into an activation table as a process which can be performed. The above-mentioned process which can be performed is performed in order of a task number, and degree treatment process is registered into the above-mentioned activation table with reference to the above-mentioned sequence information as a process which can be performed based on the activation result of the above-mentioned process. Registration of the above-mentioned process which can be performed, Activation of this process is repeated and is performed.
[0042] Moreover, the program execution approach concerning claim 21 The sequence information with which gave the task number to the task and the transition place from the relation between each task name and a sequence, the relation between the above-mentioned sequence and an event, two or more treatment processes contained in the above-mentioned sequence, and this treatment process was described to be, The activation table on which the process which can be performed is registered for every task to which this process belongs, It has an event queue for every task number. A task A task number, It is the program execution approach which carries out sequential execution of the process which was registered into the above-mentioned activation table while carrying out scheduling using the index, and which can be performed. When a registration event is in the event queue of the same task number as a current index and there is no process of this task number which can be performed in the above-mentioned activation table The 1st step which deletes the above-mentioned registration event and registers the head process of the sequence corresponding to this event into the above-mentioned activation table with reference to the above-mentioned sequence information as a process of the above-

mentioned task number which can be performed, When the process of the same task number as a current index which can be performed is registered into the above-mentioned activation table The 2nd step deleted from the above-mentioned activation table while performing this process, When there is degree treatment process with reference to the above-mentioned sequence information based on the activation result of the above-mentioned process It has the 3rd step which registers this following treatment process into the above-mentioned activation table as a process which can be performed, and the 4th step which shifts an index, and the above 1st thru/or the 4th step are repeated.

[0043] Moreover, the program execution approach concerning claim 22 In the program execution approach according to claim 21 at the 1st step of the above When a registration event is in the event queue of the same task number as a current index and the process of this task number which can be performed is shown in the above-mentioned activation table It can set up whether it replaces with this process that can be performed and the head process of the sequence corresponding to the head event of the above-mentioned event queue is registered into the above-mentioned activation table by transition authorization of the above-mentioned process which can be performed.

[0044] Moreover, the program execution approach concerning claim 23 In the program execution approach according to claim 21 or 22 When registering into the above-mentioned activation table the process which can be performed, the latency time corresponding to this process is also registered. At the time of activation of this process, when the above-mentioned latency time is not 0, this process is not performed, but it has the step which updates the latency time of the above-mentioned activation table after the 4th step of the above.

[0045] Moreover, in the program execution approach of any one publication of claim 20 thru/or claim 23, the program execution approach concerning claim 24 restricts the number of processes performed by one processing, and performs the above-mentioned processing periodically.

[0046] Moreover, the program execution approach concerning claim 25 restricts the number of processes performed by one processing for every group of a task in the program execution approach according to claim 24.

[0047] Moreover, the program execution approach concerning claim 26 restricts the number of processes performed by one processing for every drive classification of a task in the program execution approach according to claim 24.

[0048] Moreover, in the program execution approach of any one publication of claim 24 thru/or claim 26, in the above-mentioned processing, the program execution approach concerning claim 27 ends processing of the time, when the process of the total task number of the above-mentioned activation table which can be performed is checked.

[0049] Moreover, in the program execution approach of any one publication of claim 17 thru/or claim 27, the above-mentioned sequence information of the program execution approach concerning claim 28 is an executable program.

[0050] Moreover, the program execution approach concerning claim 29 is the executable program by which the above-mentioned sequence information was recorded on the record medium in the program execution approach of any one publication of claim 17 thru/or claim 28.

[0051] Moreover, in the program execution approach of any one publication of claim 17 thru/or claim 29, the above-mentioned sequence information is defined for the program execution approach concerning claim 30 by the table of a layered structure.

[0052] Moreover, the program execution equipment concerning claim 31 is equipped with the program execution section which performs sequence control by the program read the account of a top with the reading section which reads the program written in the record medium.

[0053] Moreover, the program execution equipment concerning claim 32 includes the sequence information two or more treatment processes by which the above-mentioned program is included in a sequence, and the transition place from this treatment process were described to be in program execution equipment according to claim 31.

[0054] Moreover, the program execution equipment concerning claim 33 In program execution equipment according to claim 32 the above-mentioned program execution section The activation table on which the process which can be performed is registered, and the process registration section which registers into the above-mentioned activation table the process which can be performed with reference to the above-mentioned program, The above-mentioned process which can be performed is performed, and it judges whether the activation result is normal, and has the processes run section which makes a process [degree] process or a

transition place process register into the above-mentioned process registration section as a process which can be performed.

[0055] Moreover, as for the program execution equipment concerning claim 34, two or more treatment processes by which the above-mentioned program is included in the relation between each task name and a sequence, the relation between the above-mentioned sequence and an event, and the above-mentioned sequence in program execution equipment according to claim 31, and the transition place from this treatment process include the described sequence information.

[0056] Moreover, in program execution equipment according to claim 34, as for the program execution equipment concerning claim 35, the above-mentioned program execution section also performs scheduling of a task with activation of sequence control using a task number and an index.

[0057] Moreover, the program execution equipment concerning claim 36 In program execution equipment according to claim 34 or 35 the above-mentioned program execution section The activation table on which the task number was given for the process which can be performed and which is registered for every task, The process registration section which registers into the above-mentioned activation table the process which can be performed with reference to the above-mentioned program, The event queue table on which the generated event has the event queue with which this event belongs, and which is registered for every task, The event analysis section which will register this event into the event queue corresponding to the task to which this event belongs with reference to the above-mentioned program if an event occurs, An instruction is given to the process registration section so that it may register with the above-mentioned activation table by making each process of the sequence control corresponding to a registration event into the process which can be performed. It has the sequence activation section which performs the process of the same task number as this index which can be performed, shifting an index.

[0058] Moreover, as for the above-mentioned program, the program execution equipment concerning claim 37 is making the layered structure in the program execution equipment of any one publication of claim 31 thru/or claim 36.

[0059] Moreover, the sequence control approach concerning claim 38 It has data with which two or more treatment processes contained in a sequence and the transition place from this treatment process were described. Are the sequence control approach of performing the above-mentioned sequence control, and the contents of the above-mentioned data are read. It changes into a program, and stores in the storage section, sequential execution of each process of sequence control is carried out with reference to the program of the above-mentioned storage section, and degree treatment process is performed with reference to the above-mentioned program based on the activation result of the above-mentioned process.

[0060] Moreover, the sequence control approach concerning claim 39 The data with which the task number was given to the task and the transition place from the relation between each task name and a sequence, the relation between the above-mentioned sequence and an event, two or more treatment processes contained in the above-mentioned sequence, and this treatment process was described, The activation table on which the process which can be performed is registered for every task to which this process belongs, It has an event queue for every task number. A task A task number, The 1st step which is the sequence control approach which carries out sequential execution of the above-mentioned activation ****, carrying out scheduling using an index, reads the contents of the above-mentioned data, changes into a program, and is stored in the storage section, When a registration event is in the event queue of the same task number as a current index and there is no process of this task number which can be performed in the above-mentioned activation table The 2nd step which deletes the above-mentioned registration event and registers the head process of the sequence corresponding to this event into the above-mentioned activation table with reference to the above-mentioned sequence information as a process of the above-mentioned task number which can be performed, When the process of the same task number as a current index which can be performed is registered into the above-mentioned activation table The 3rd step deleted from the above-mentioned activation table while performing this process, When there is degree treatment process with reference to the above-mentioned sequence information based on the activation result of the above-mentioned process It has the 4th step which registers this following treatment process into the above-mentioned activation table as a process which can be performed, and the 5th step which shifts an index, and the above 2nd thru/or the 5th step are repeated.

[0061] Moreover, in the sequence control approach according to claim 38 or 39, the above-mentioned data are defined for the sequence control approach concerning claim 40 by the table of a layered structure.

[0062] Moreover, the sequence control equipment concerning claim 41 The data division two or more treatment

processes contained in a sequence and the transition place from this treatment process were described to be, The program generation section which reads the contents of the above-mentioned data division and is changed into a program, With reference to the above-mentioned program stored in the storage section in which the above-mentioned program is stored, and the above-mentioned storage section, sequential execution of each process of sequence control is carried out. Based on the activation result of the above-mentioned process, it has the program execution section which performs degree treatment process with reference to the above-mentioned program.

[0063] Moreover, the sequence control equipment concerning claim 42 In sequence control equipment according to claim 41 the above-mentioned program execution section The activation table on which the process which can be performed is registered, and the process registration section which registers into the above-mentioned activation table the process which can be performed with reference to the program of the above-mentioned storage section, The above-mentioned process which can be performed is performed, and the activation result judges whether it is normal, and is equipped with the processes run section which makes a process [degree] process or a transition place process register into the above-mentioned process registration section as a process which can be performed.

[0064] Moreover, the sequence control equipment concerning claim 43 The relation between each task name and a sequence, and the relation between the above-mentioned sequence and an event, The data division two or more treatment processes contained in the above-mentioned sequence and the transition place from this treatment process were described to be, The contents of the above-mentioned data division are read and the above-mentioned program stored in the program generation section changed into a program, the storage section in which the above-mentioned program is stored, and the above-mentioned storage section is referred to. With activation of sequence control It has the program execution section which also performs scheduling of a task using a task number and an index.

[0065] Moreover, the sequence control equipment concerning claim 44 In sequence control equipment according to claim 43 the above-mentioned program execution section The activation table on which the task number was given for the process which can be performed and which is registered for every task, The process registration section which registers into the above-mentioned activation table the process which can be performed with reference to the above-mentioned program, The event queue table on which the generated event has the event queue with which this event belongs, and which is registered for every task, The event analysis section which will register this event into the event queue corresponding to the task to which this event belongs with reference to the above-mentioned program if an event occurs, An instruction is given to the process registration section so that it may register with the above-mentioned activation table by making each process of the sequence control corresponding to a registration event into the process which can be performed. It has the sequence activation section which performs the process of the same task number as this index which can be performed, shifting an index.

[0066] Moreover, in the sequence control equipment of any one publication of claim 41 thru/or claim 44, the above-mentioned data division are defined for the sequence control equipment concerning claim 45 by the table of a layered structure.

[0067] Moreover, the task-scheduling program documentation medium concerning claim 46 has the task activation table on which the task which can be performed is registered, and records the program performed in order of the task number to which the above-mentioned task which can be performed was given by the task in the task-scheduling program documentation medium which recorded the program which controls the execution sequence of the above-mentioned task which can be performed.

[0068] Moreover, the program generator record medium concerning claim 47 records the program which reads the data with which two or more treatment processes contained in a sequence and the transition place from this treatment process were described, the program which changes the contents of the above-mentioned data into a program source code, and the program which compiles and links the above-mentioned program source code, and generates an executable program.

[0069] Moreover, the program execution program documentation medium concerning claim 48 It has the sequence information two or more treatment processes contained in a sequence and the transition place from this treatment process were described to be. The program which is the program execution program documentation medium which recorded the program which performs sequence control, and carries out sequential execution of the process of sequence control with reference to the above-mentioned sequence information, Based on the activation result of the above-mentioned process, the program which performs degree

treatment process with reference to the above-mentioned sequence information is recorded.
[0070] Moreover, the program execution program documentation medium concerning claim 49 The sequence information with which gave the task number to the task and the transition place from the relation between each task name and a sequence, the relation between the above-mentioned sequence and an event, two or more treatment processes contained in the above-mentioned sequence, and this treatment process was described to be, The activation table on which the process which can be performed is registered for every task to which this process belongs, It has an event queue for every task number. A task A task number, It is the program execution program documentation medium which recorded the program which carries out sequential execution of the above-mentioned process which can be performed while carrying out scheduling using an index. When a registration event is in the event queue of the same task number as a current index and there is no process of this task number which can be performed in the above-mentioned activation table The 1st step which deletes the above-mentioned registration event and registers the head process of the sequence corresponding to this event into the above-mentioned activation table with reference to the above-mentioned sequence information as a process of the above-mentioned task number which can be performed, When the process of the same task number as a current index which can be performed is registered into the above-mentioned activation table The 2nd step deleted from the above-mentioned activation table while performing this process, When there is degree treatment process with reference to the above-mentioned sequence information based on the activation result of the above-mentioned process It has the 3rd step which registers this following treatment process into the above-mentioned activation table as a process which can be performed, and the 4th step which shifts an index, and the program which repeats the above 1st thru/or the 4th step is recorded.
[0071] Moreover, the sequence control program documentation medium concerning claim 50 It has data with which two or more treatment processes contained in a sequence and the transition place from this treatment process were described. The program which is the sequence control program documentation medium which recorded the program which performs the above-mentioned sequence control, reads the contents of the above-mentioned data, changes into a program, and is stored in the storage section, The program which carries out sequential execution of each process of sequence control with reference to the program of the above-mentioned storage section, and the program which performs degree treatment process with reference to the above-mentioned program based on the activation result of the above-mentioned process are recorded.
[0072] Moreover, the record medium concerning claim 51 is a record medium which recorded as data two or more treatment processes contained in a sequence, and the transition place from this treatment process, and is equipped with the 1st field which comes to record two or more above-mentioned treatment processes, and the 2nd field which relates with each of two or more above-mentioned treatment processes, and comes to record an unit or two or more transition place processes.
[0073] Moreover, the record medium concerning claim 52 is equipped with the 3rd field where the above-mentioned record medium comes to record the latency time of the above-mentioned process in relation to the above-mentioned process further in a record medium according to claim 51.
[0074] Moreover, the event corresponding to a task name and this task name in the record medium concerning claim 53, The sequence corresponding to this event, and two or more treatment processes which can be set to this sequence, The 1st field which is the record medium which recorded the transition place from this treatment process as data, and comes to record the above-mentioned task name, The 2nd field which relates with the above-mentioned task name and comes to record an unit or two or more events, It relates with each of the above-mentioned event, and has the 3rd field which comes to record a sequence. The 3rd field of the above Furthermore, the process record section which comes to record the process which relates with each of the above-mentioned sequence and is included in the above-mentioned sequence, the transition place process record section which the above-mentioned process is alike, respectively, and relates and comes to record an unit or two or more transition place processes -- since -- it is constituted.
[0075]
[Embodiment of the Invention] (Gestalt 1 of operation)
[0076] Hereafter, the task-scheduling equipment by the gestalt 1 of operation of this invention and an approach are explained, referring to a drawing. Drawing 1 is the block diagram showing the configuration of the task-scheduling equipment of the gestalt 1 of this operation. Drawing 2 R> 2 is drawing showing a task activation table. Drawing 3 is a flow chart which shows the task-scheduling approach of the gestalt 1 this operation. The task-scheduling equipment 100 shown in drawing 1 is equipped with the task registration section 101 and the task-scheduling section 102. Furthermore, the task-scheduling section 102 is equipped with the task activation

table 103 and the task activation section 104. The task registration section 101 detects the activate request of a task, and registers this task into the task activation table 103 in the task-scheduling section 102. The task activation table 103 is a table which registers temporarily the task which should be performed, for example, is shown by drawing 2 (a) and (b). The above-mentioned task activation table 103 has the magnitude of only the number of tasks, and is making each task and a task number correspond to one to one. The task activation section 104 carries out scheduling of the activation of a task using INDEX and a task number with reference to the task registered into the above-mentioned task activation table 103. Above INDEX is a variable which takes one value of the total task numbers.

[0077] Next, actuation of the task-scheduling equipment 100 of the gestalt 1 of this operation is explained using drawing 2 and drawing 3 . In the gestalt 1 of this operation, CPU with which the task activation section 104 is equipped and which is not illustrated presupposes as an example that processing is performed every 5ms and a maximum of three tasks can be performed by one processing. First, before the processing in CPU which is not illustrated is started, suppose that the task activate request of task A-E occurred. Then, the task registration section 101 detects the activate request of each above-mentioned task, and registers each task into the task of the task activation table 103 which can be performed. The task activation table 103 on which the task was registered has become like drawing 2 (a). In addition, since explanation is easy, suppose that generating of the activate request of each of this task and the registration to the task activation table 103 were happened only once.

[0078] Next, the task-scheduling approach of the task-scheduling section 102 is explained using the flow chart of drawing 3 . CPU which the task activation section 104 does not illustrate -- time amount =0ms -- the 1st processing -- starting -- first -- the number of activation tasks -- it is referred to as =0 and loop count =0 (step S301).

[0079] Next, 1 is added to loop count and it is referred to as loop count =1 (step S302). And whether with reference to the task activation table 103 of drawing 2 (a), the task of the task number corresponding to the present INDEX which can be performed is registered judges the task activation section 104 (step S303), and if registered, it will take out this task, will delete it from the task activation table 103, and will be performed (steps S304 and S305). Here, at the time of processing initiation, only once, INDEX is initialized and it is referred to as INDEX=0. The task whose task number is 0 is Task A, and since Task A is registered into the task which can be performed, Task A is deleted from the task activation table 103, and is performed (step S 304 305). and the number of activation tasks -- 1 -- adding -- the number of activation tasks -- it is referred to as =1 (step S306). In addition, in step S303, when the task of the task number corresponding to current INDEX which can be performed is not registered, step S304 - step S306 are skipped, and it progresses to step S307.

[0080] Next, at step S307, 1 is added to INDEX and it is referred to as INDEX=1. And INDEX judges whether it is more than the number of task activation tables (step S308). if it is more than the number of the task activation tables 103 -- INDEX=0 -- replacing (step S309) -- since it is INDEX=1 this time, replacement of INDEX is not performed.

[0081] Next, it judges whether the number of activation tasks is over the number of the maximum activation (three pieces) (step S310). The number of activation tasks = by 1, the number of the maximum activation is exceeded, since ****, it progresses to step S311 and loop count still judges whether it is under the number of activation tables. Since loop count (1 time) is under the number of task activation tables (= 5), it repeats processing of return and the above flow to step S302. In addition, also when, as for the check of whether this loop count is under the number of activation tables, the task is not registered into the task activation table 103, the task activation section 104 is decision required in order to end processing of that time, when seeing briefly the column of the task of the task activation table 103 which can be performed. If there is no decision of step S311, when there will be no registration of a task in the task activation table 103, the task activation section 104 checks the task activation table 103 eternally, and cannot end processing of the time.

[0082] thus -- if Task A, Task B, and Task C are performed -- the number of activation tasks -- it is set to =3 and ended by decision of step S310. At this time, Task A, Task B, and Task C are deleted from the task activation table 103.

[0083] It is INDEX=3 when the 1st processing is completed. Moreover, the task activation table 103 when the 2nd processing is started (time amount = 5ms) has become like drawing 2 (b). In addition, in drawing 2 (b), it is shown that the place whose column of the task which can be performed is "-" does not have the task registered. In the 2nd processing by CPU which the task activation section 104 does not illustrate, it performs from Task D, Task D and Task E are performed, and the processing which is the 2nd time is ended. The task activation

section 104 merely performs only the check of the task which can be performed every 5ms until there is registration of the new task to the task activation table 103, since the task activation table 103 is the state of the sky after that.

[0084] Thus, without according to the task-scheduling approach by the gestalt 1 of this operation, constituting a complicated program and performing fine control by having considered as the approach of performing the task of the task number corresponding to this INDEX, giving a task number to a task and shifting INDEX in order, it is possible to carry out scheduling of the task, and it can also prevent that a specific task remains not performing. Moreover, since the configuration of a program becomes easy, the size of a program also becomes small. Furthermore, since the number of the maximum tasks performed by one processing is controllable by step S310 of drawing 3 R> 3 according to the capacity of CPU, activation of a task cannot be completed within 1 time of the processing time, but the situation that the starting timing of a resident task and a fixed-cycle task is not securable can also be avoided.

[0085] In addition, although the number of task activation tables is one, and all tasks shall be registered into the above-mentioned task activation table by the task-scheduling equipment of the gestalt 1 of this operation, and the approach when the activate request of a task occurs It is also possible to set up the number of activation tasks which this is an example, for example, divides all tasks into the group of two or more tasks, and is equipped with a task activation table for this every group, and is performed by one task executive operation for every group. At this time, from initiation of the flow chart of drawing 3 to termination will be repeated for every group of each task in one task executive operation. Therefore, when the number of the groups of the above-mentioned task is three, it is necessary to prepare three kinds of INDEX(s) like INDEX (1), INDEX (2), and INDEX (3) for every group. In addition to the same effectiveness as the gestalt 1 of this operation, a task with the above-mentioned high priority has the effectiveness which comes to be performed more certainly by assigning more numbers of activation tasks to the group of a task with a still higher priority by having a task activation table for every specific group of a task as mentioned above, and setting up the number of activation tasks for every above-mentioned group. Moreover, it is also possible to carry out the group division of the task by drive classification of a task, and there is effectiveness performed by being stabilized at this time, without a fixed-cycle task and a resident task being barred by the activation of an event task started irregularly.

[0086] Moreover, although [the gestalt 1 of this operation / INDEX] a task number and INDEX are continuous integers which begin from 0, and INDEX is shifted when only 1 makes this INDEX increase The figure which changes with some fixed regulations, such as a continuous integer which this is an example, for example, begins from 1, or continuous even number which begins from 0, and not overlapping, Or if it is a notation etc., it will be possible to also use them as a task number and INDEX and for it to be possible, and to decrease INDEX further, or to also make it change according to a certain fixed regulation, and the same effectiveness will be acquired.

[0087] Moreover, with the gestalt 1 of this operation, although processing which shifts one INDEX is performed after activation of a task, it is also possible to perform the processing which this is an example, for example, shifts one INDEX in drawing 3 , step S308, and step S309 in front of the activation S303 of a task, i.e., a step, etc., and the same effectiveness is acquired.

[0088] Moreover, with the gestalt 1 of this operation, although processing spacing of task activation was set to 5ms, this is an example, no matter it may be what processing spacing, scheduling of a task can be realized and equivalent effectiveness is acquired.

[0089] Moreover, although [the gestalt 1 of this operation] task executive operation is started periodically, even if such, the scheduling of the task using a task number and INDEX is possible [this is an example, for example, it is also possible to perform task executive operation continuously, and].

[0090] Moreover, although [the gestalt 1 of this operation] a task is taken out from a task activation table, is deleted and is performed in the flow chart of drawing 3 , this is an example, and after it performs a task, it may delete this task from the above-mentioned task activation table.

[0091] (Gestalt 2 of operation)

[0092] Hereafter, the sequence control equipment by the gestalt 2 of operation of this invention and an approach are explained, referring to a drawing. Drawing 4 is the block diagram showing the configuration of the sequence control equipment 400 by the gestalt 2 of this operation. Drawing 5 is drawing showing the structure of data, and drawing 6 is drawing showing a sequence table and a transition table. Drawing 7 is drawing showing an example of an activation table, a sequence table, and a transition table. The sequence control equipment 400 shown in drawing 4 is equipped with data division 401, the program generation section 402, the storage section

403, and the program execution section 404. Moreover, data division 401 are equipped with a sequence table 405 and the transition table 406. The storage section 403 is equipped with the source code storing section 412, the executive program storing section 413, and memory 414. The program generation section 402 is equipped with the interpretation section 407 and the generation section 408. The program execution section 404 is equipped with the process registration section 409, the activation table 410, and the processes run section 411.
[0093] It is data which have a layered structure as indicated to be a sequence table 405 and the transition table 406 to drawing 5 . When expressed as a table, the transition table 406 is shown like drawing 6 (b) like drawing 6 (a) in a sequence table 405. In the sequence table of drawing 6 (a), a process is processing which CPU performs and the latency time is time amount for which it waits until it performs the above-mentioned process. Moreover, the transition table of drawing 6 (b) is a table showing the transition place process which changes when the activation result of the above-mentioned process is not normal.
[0094] The interpretation section 407 interprets to what kind of process the data indicated there correspond with reference to the data indicated by data division 401. The generation section 408 describes the contents of the data interpreted in the above-mentioned interpretation section 407 with programming language, and stores the generated program source code in the source code storing section 412 of the storage section 403. The program source code stored in the source code storing section 412 of the storage section 403 is compiled and linked, and serves as an executable program, and this executable program is stored in the executive program storing section 413. The above-mentioned executable program is stored in memory 414 at the time of activation. The process registration section 409 registers a process into the activation table 410 with reference to the above-mentioned sequence table 405 stored in the above-mentioned memory 414, and the above-mentioned transition table 406. The activation table 410 is equipped with the cel which can write in one process which can be performed, and the cel which can write in the latency time corresponding to this process as shown by drawing 7 (a). The processes run section 411 performs the process registered into the process which can be performed with reference to the above-mentioned activation table 410. Moreover, reception and the process registration section 409 are made to register a new process into the activation table 410 for the activation result of the above-mentioned process.
[0095] Next, actuation of the sequence control equipment by the gestalt 2 of this operation and the sequence control approach are explained using drawing 8 . Drawing 8 is a flow chart which shows the sequence control approach. Hereafter, the same sequence control as the program generation method of the conventional example is explained as an example. First, the process of the same sequence control as the above-mentioned conventional example etc. is written down in the sequence table 405 and the transition table 406 of data division 401. For example, using spreadsheet software etc., as shown in drawing 7 (b), (c), (d), and (e), each above-mentioned table is created by entering a concrete process name, the latency time, a transition place, etc. in each cel.
[0096] Next, the interpretation section 407 of the program generation section 402 interprets what kind of processes they are concretely with reference to a process name, a transition place name, etc. which were indicated by each table of the above-mentioned data division 401. for example, a spreadsheet -- a character string like drawing 7 (b) indicated in the soft table etc. is read, and it interprets of what kind of contents the "processing A" etc. is processing concretely. And the interpretation result is told to the generation section 408. The generation section 408 describes the table data of the layered structure indicated by data division 401 with programming language based on the interpretation result from the above-mentioned interpretation section 407, and stores the generated program source code in the source code storing section 412 of the storage section 403. Therefore, the program itself currently recorded on the storage section 403 is only what described the data of the layered structure shown in drawing 5 with programming language. The above-mentioned program source code is compiled and linked, serves as an executable program, and is stored in the executive program storing section 413. This executable program is stored in memory 414 at the time of activation. The program execution section 404 performs sequence control, referring to the program stored in the above-mentioned memory 414.
[0097] Hereafter, the above-mentioned sequence control is explained using the flow chart of drawing 8 . First, an instruction is issued so that the processes run section 411 may register a head process and its latency time into the process of the activation table 410 which can be performed at the process registration section 409. Then, the process registration section 409 reads the head process and the latency time of a sequence table 405 with reference to memory 414, and registers them into the activation table 410 (step S801). The activation table 410 on which the processing A which is the head process of the sequence table of drawing 7 (b) was registered is shown by drawing 7 (a).

[0098] Next, the latency time of the process in which the processes run section 411 was registered into the process of the activation table 410 which can be performed = it judges in 0 (S802). Since the latency time of Processing A is 0, Processing A is taken out and deleted from the activation table 410 (step S804).
[0099] Next, it progresses to step S805 and processing A taken out from the activation table 410 is performed. The activation result of Processing A is returned to the processes run section 411. As for the processes run section 411, this activation result judges whether it is normal (step S806). Although the column of a transition table is "-" about Processing A, it is shown that this does not have a setup of a transition table and all activation results become normal. Therefore, when whether a process [degree] process is in the process registration section 409 makes the processes run section 411 judge and there is a process [degree] process, it makes this process register into the activation table 410. the process registration section 409 has a process [degree] process "conditional judgment 1" in a sequence table with reference to memory 414 -- checking (step S807) -- this -- a process [degree] process and its latency time are registered into the activation table 410 (step S808).
[0100] And the latency time corresponding to the conditional judgment 1 by which return and the processes run section 411 were registered into step S802 by the process of the activation table 410 which can be performed = it judges whether it is 0. Since the latency time of conditional judgment 1 is 20ms, waiting for time amount is carried out for 20ms (step S803), and conditional judgment 1 is taken out from the activation table 410, is deleted, and is performed (steps S804 and S805).
[0101] The activation result of the above-mentioned process is returned to the processes run section 411, and this activation result judges whether it is normal (step S806). In activation of the above-mentioned conditional judgment 1, when an activation result becomes normal when fulfilling conditions 1, and you do not fulfill conditions 1, suppose that an activation result is not normal. When an activation result is not normal, the processes run section 411 issues an instruction so that a transition place process may be registered into the activation table 410 at the process registration section 409 with reference to the transition table corresponding to the above-mentioned conditional judgment 1. The process registration section 409 registers into the activation table 410 the conditional judgment 2 whose process number is 3 with reference to the transition table A of drawing 7 (c) of the Records Department 403. And the processing from step S802 is repeated again.
[0102] In the above-mentioned step S806, when it judges that the activation result of the above-mentioned conditional judgment 1 has the normal processes run section 411, an instruction is issued so that a process [degree] process may be registered into the activation table 410 at the process registration section 409, and the above-mentioned process registration section 409 registers into the activation table 410 the processing 1 which is a process [degree] process. And the processing from step S802 is repeated again. Thus, if the processes run section 411 performs even processing C whose process number is 7, since there is no process [degree] process, it will be that it is ended.
[0103] Thus, according to the sequence control equipment in the gestalt 2 of this operation, and the approach, the program itself is only data of a layered structure, it is having had the program execution section 404 which can perform the program of only the data, and can perform the C tense control of the program of only the above-mentioned data. Moreover, as mentioned above, since it can respond to the addition of the process to this program, deletion, modification, etc. easily since a program is only data, and a change of the transition place as conditional branching can also be easily made only by modification of the contents of the transition table, the adaptation range is wide and the high sequence control equipment of description capacity and an approach can be offered. For example, what is necessary is just to add Processing D to the sequence table of drawing 7 (b) as a sequence number 8 in the example of drawing 7 to add Processing D after Processing C.
[0104] Furthermore, a program will be only data, and since conditional branching etc. is not included in a program, even if program capacity serves as necessary minimum, and the capacity of a program decreases compared with the conventional thing and it adds processing, the increment in the capacity of a program will require only a part for the data of the processing. Therefore, versatility is very high and serves as sequence control equipment which can respond to various sequence control flexibly, and an approach. as mentioned above, compared with the conventional thing, the size effectiveness of a program is boiled markedly and is improving.
[0105] In addition, although the transition place process specified on a transition table was set to one in the program generation method of the gestalt 2 of this operation, this is an example, in decision of step S806 of the flow chart of drawing 8 , when the activation result of a process is not normal, it is also possible to choose either from two or more transition places, and the same effectiveness is acquired. For example, in the sequence table of drawing 7 (b), change the transition table A into transition table A' of drawing 7 (f), and it sets to

activation of the conditional judgment 1 of a sequence number 1. Although a result presupposes that it is normal and is not applied to conditions 1, when applied to conditions 1, when applied to conditions 2, Although it is not applied to conditions 1 and conditions 2, when applied [to conditions 3] and not applied to conditions 1, conditions 2, and conditions 3, it can also set up, respectively so that it may change in the process of the transition number 1, the transition number 2, and the transition number 3. Thus, a transition place can be easily changed only by modification of a transition table.

[0106] Moreover, although [the gestalt 2 of this operation / data division 401 the program generation section 402, the storage section 403, and the program execution section 404] all are contained in one sequence control equipment 400 This is an example and the program recorded on the program generation equipment 900 which generates a program, the record medium 903 which records the program by which generation was carried out [above-mentioned], and the above-mentioned record medium 903 is read like drawing 9 R> 9. It is also possible to divide into the program execution equipment 901 which performs sequence control. In addition, let the approach of performing sequence control for the approach of generating a program with the above-mentioned program generation equipment 900 with a program generation method and the above-mentioned program execution equipment 901 be the program execution approach. Here, in drawing 9 , the same sign as drawing 4 shows the same thing as the sequence control equipment of the gestalt 2 of this operation, and those explanation is omitted. The write-in section 902 writes the program which the program generation section 402 generated in a record medium 903. An executable program is written in the above-mentioned record medium 903. Moreover, the reading section 904 reads the executable program written in the record medium 903. The read program is stored in the memory which the above-mentioned reading section 904 does not illustrate.

[0107] Since only program execution equipment 901 can be further used independently on the same effectiveness as the gestalt 2 of this operation by dividing the sequence control equipment 400 of the gestalt 2 of this operation into program generation equipment 900 like drawing 9 , program execution equipment 901, and a record medium 903, the part and the equipment which do not contain program generation equipment 900 compared with sequence control equipment 400 can be miniaturized. Moreover, since it is also possible to realize sequence control with two or more program execution equipments 901 even if there is only single program generation equipment 900, the cost can be cut down compared with the case where two or more sequence control equipments 400 are used. Furthermore, the sequence control of program execution equipment 901 can also be easily changed only by modification of a record medium 903 by recording the data program of the sequence control of two or more classes on the record medium 903 with program generation equipment 900 in advance.

[0108] Moreover, with the gestalt 2 of this operation, the process registration section 409 accesses the executable program stored in the memory 414 of the storage section 403. Although the process included in this executable program shall be registered into the activation table 410 and the processes run section 411 shall perform the process by which registration was carried out [above-mentioned] This is an example, it is also possible to perform sequence control, not having a component which registers a process temporarily as shown in the above-mentioned activation table 410, but accessing the memory 414 of the storage section 403 directly, and the same effectiveness is acquired.

[0109] Moreover, with the gestalt 2 of this operation, although the number of sequence tables was set to one, this is an example and preparing two or more sequence tables, or also considering as the sequence of a layered structure by [of a sequence table] preparing the subsequence table of the hierarchy under one more etc. further and the effectiveness that it is possible and the description capacity other than the same effectiveness as the gestalt 2 of this operation becomes high further are acquired.

[0110] Moreover, with the gestalt 2 of this operation, although the latency time corresponding to a process was set up, this is an example, for example, it is also possible to perform time amount waiting of only the necessary latency time as one process occasionally whose latency time is the need, without setting up the latency time, and the same effectiveness is acquired.

[0111] Moreover, although [the gestalt 2 of this operation] a process is taken out from an activation table, is deleted and is performed in the flow chart of drawing 8 , this is an example, and after it performs a process, it may delete this process from the above-mentioned activation table. Or overwriting is also possible, in case the process to delete is skipped and a process [degree] process or a transition place process is registered into an activation table.

[0112] Moreover, with the gestalt 2 of this operation, a process [degree] process is an unit, although the transition place process should be chosen as the transition table based on the activation result from the multiple

processes of a publication, this is an example, it is also possible to choose a process [degree] process when the activation result of a process is normal from multiple processes, and the same effectiveness is acquired.

[0113] (Gestalt 3 of operation)

[0114] The sequence control equipment by the gestalt 3 of operation of this invention and an approach perform sequence control included in the above-mentioned task, carrying out scheduling of two or more tasks by INDEX. Hereafter, the sequence control equipment of the gestalt 3 of this operation and an approach are explained, referring to a drawing. Drawing 10 is the block diagram showing the configuration of the sequence control equipment of the gestalt 3 of this operation. Sequence control equipment 1000 is equipped with data division 1001, the program generation section 1002, the storage section 1003, and the program execution section 1004. Moreover, data division 1001 are equipped with the task table 1005, the event table 1006, a sequence table 1007, the subsequence table 1008, and the transition table 1009. The program generation section 1002 is equipped with the interpretation section 1010 and the generation section 1011. The storage section 1003 is equipped with the source code storing section 1017, the executive program storing section 1018, and memory 1019. The program execution section 1004 is equipped with the event analysis section 1012, the event queue table 1013, the sequence activation section 1014, the process registration section 1015, and the activation table 1016.

[0115] Drawing 11 is drawing having shown the data-hierarchy structure of the task table 1005, the event table 1006, a sequence table 1007, and the subsequence transition table 1008 and 1009. Drawing 12 is drawing having shown each above-mentioned table. The task table 1005 of drawing 12 (a) indicates correspondence with the event table 1006 to be each task. The event table 1006 of drawing 12 (b) shows correspondence with the event belonging to this event table 1006, and event reception conditions and a sequence table 1007.

[0116] When the above-mentioned event reception conditions are "with no queue", in case the generated event is registered into the event queue corresponding to the above-mentioned event of the event queue table 1013, it registers, after deleting the event by which the queuing was carried out by then, and the generated event is registered into the last edge of the above-mentioned event queue when the above-mentioned event reception conditions are "with a queue." Therefore, the urgent event reception conditions of an event presuppose "with no queue", and let the event reception conditions of the low event of an urgency be "those with a queue."

[0117] The sequence table 1007 of drawing 12 (c) has further two or more subsequence tables 1008. The subsequence table 1008 of drawing 12 (d) has multiple processes further, and the latency time, transition authorization, and the transition table 1009 support this process. It is the above-mentioned process which CPU actually performs in sequence control. The above-mentioned latency time is the latency time until activation of the process is started.

[0118] Moreover, when the above-mentioned transition authorization is "authorization" and an event is in an event queue, the sequence under activation is ended. Register the process corresponding to the above-mentioned event into the activation table 1016, perform it, and when the above-mentioned transition authorization is "prohibition" or [that the process of "authorization" is registered into the activation table 1016 for transition authorization in the sequence under activation even when an event is in an event queue] -- or The sequence corresponding to the above-mentioned event cannot be started until all the processes of the above-mentioned sequence are completed.

[0119] The transition table 1009 of drawing 12 (e) shows the transition place process which changes when the activation result of the above-mentioned process is not normal by the sequence table 1007, the subsequence number, and the process number. The interpretation section 1010 interprets whether with reference to the data indicated by data division 1001, the contents indicated there are equivalent to what kind of process. The generation section 1011 describes the contents of the data interpreted in the above-mentioned interpretation section 1010 with programming language, and stores the generated program source code in the source code storing section 1017 of the storage section 1003. The program source code stored in the source code storing section 1017 of the storage section 1003 is compiled and linked, and serves as an executable program, and this executable program is stored in the executive program storing section 1018. The above-mentioned executable program is stored in memory 1019 at the time of activation. The event analysis section 1012 detects that the event occurred, and carries out the queuing of the event concerned to the event queue corresponding to the above-mentioned event of the event queue table 1013.

[0120] Each event queue of the event queue table 1013 stores the generated event. Drawing 13 (a) is drawing having shown this event queue table 1013. Each task has an event queue corresponding to this task. That is, an event queue has only the class of task. An event queue is the DS of FIFO (First In First Out), the generated

event is registered behind the registered event of the event queue corresponding to this event, and the event registered is taken out and deleted from the head side. If a top event is deleted, the event registered into the degree will shift to a head side by one.

[0121] The activation table 1016 has the magnitude of only the number of tasks, and the process which each task and a task number should be made to correspond to one to one, and then should perform them and the latency time corresponding to it, and transition authorization are stored temporarily. Drawing 13 (b) is drawing showing the above-mentioned activation table 1016. With reference to the above-mentioned event queue activation table 1013 and 1016, the sequence activation section 1014 carries out scheduling of two or more tasks using the above-mentioned task number and INDEX, and performs a process. Above INDEX is a variable which takes one value of the total task numbers. The process registration section 1015 registers a process into the activation table 1016, referring to the data program of memory 1019 based on the instruction from the above-mentioned sequence activation section 1014.

[0122] Next, actuation of the sequence control equipment 1000 of the gestalt 3 of this operation and the sequence control approach are explained using drawing 16 , drawing 17 , and drawing 18 . Drawing 16 is the flow chart which showed actuation of the event analysis section 1012. Drawing 17 is the flow chart which showed actuation of the program execution section 1004. Drawing 18 is the flow chart which showed the predefined process in the flow chart of drawing 17 "a processes run."

[0123] First, the process of the sequence control to wish, the relation between the event table 1006 and a task, etc. are entered in the task table 1005, the event table 1006, the sequence table 1007, and the subsequence transition table 1008 and 1009 of data division 1001 of a layered structure like drawing 11 . For example, each above-mentioned table is created like drawing 12 (a) - (e) using spreadsheet software etc. by entering a concrete process name, the latency time, a table name, etc. in each cel.

[0124] Next, the interpretation section 1010 of the program generation section 1002 interprets the event [what kind of process] they support concretely with reference to a process name, an event name, etc. which were indicated by each table of the above-mentioned data division 1001. And the interpretation result is told to the generation section 1011. The generation section 1011 describes the table data of the layered structure indicated by data division 1001 with programming language based on the interpretation result from the above-mentioned interpretation section 1010, and stores the generated program source code in the source code storing section 1017 of the storage section 1003. And the above-mentioned program source code is compiled and linked, serves as an executable program, and is stored in the executive program storing section 1018. The above-mentioned executable program is stored in memory 1019, and is performed. Therefore, the program itself stored in the storage section 1003 is what [only] described the data of the layered structure shown in drawing 11 by the program source code. That is, it is a data program and the executive program of the sequence control is not contained. Referring to the program recorded on the above-mentioned memory 1019, the program execution section 1004 carries out scheduling of the task using INDEX and a task number, and performs sequence control corresponding to the task.

[0125] Hereafter, actuation of sequence control equipment 1000 is explained by making control of an air-conditioner into an example. Some table data in the air-conditioner control are shown in drawing 14 . First, the table data shown in drawing 14 are changed into a program by the program generation section 1002, and presuppose after that that it was compiled and linked and was stored in memory 1019.

[0126] Next, when a certain event occurs externally, the event analysis section 1012 detects that and carries out the queuing of this event to the event queue corresponding to the above-mentioned event of the event queue table 1013 according to a fixed regulation. Actuation of the event analysis section 1012 is shown by the flow chart of drawing 16 .

[0127] The event analysis section 1012 supervises whether there is any generating of an event continuously as a resident task. And if there is generating of an event, with reference to the event table as a data program currently recorded on memory 1019, it will check first to which event table the generated event belongs. Next, with reference to a task table, correspondence with the above-mentioned event table and a task is checked (step S1601). And when it judges whether there is any registration event to the event queue corresponding to the above-mentioned task of the event queue table 1013 (step S1602) and there is no registration event in it, it progresses to step S1605 and the event generated in the above-mentioned event queue is registered.

[0128] In the example of air-conditioner control, like drawing 15 (a), the air conditioning carbon button of remote control is pushed, and suppose that the event "air conditioning" occurred. As shown in the event queue table 1013 on the right-hand side of drawing 15 (a), supposing processing has still started just and anything does

not have an event in the event queue table 1013 at this time, the event analysis section 1012 will carry out the queuing of the event "an air conditioning key" to the head of an air conditioning event queue.
[0129] In step S1602 of drawing 16 , when a registration event is in the event queue corresponding to the generated event, the reception conditions of the generated event judge "he has no queue" and "those with a queue" (step S1603). And when event reception conditions are "with no queue", all the registration events of the above-mentioned event queue are deleted, and the generated event is registered into the head of an event queue. By decision of step S1603, when event reception conditions are "with a queue", the event concerned is registered behind the event progressed and registered into step S1605. As mentioned above, since it is a resident task, the event analysis section 1012 always supervises generating of an event, and when there is generating of an event, it repeats processing from initiation of the flow chart of drawing 16 to termination.
[0130] Next, suppose that the processing in CPU which the sequence activation section 1014 does not illustrate was started. In the gestalt 3 of this operation, executive operation is started every 5ms and suppose that a maximum of three processes can be performed by one processing. if CPU which the sequence activation section 1014 does not illustrate starts processing -- first -- the number of activation processes -- it is referred to as =0 and loop count =0 (step S1701), and next, 1 is added to loop count and it is referred to as loop count =1 (step S1702). And it judges whether the event is registered into the event queue of the task number corresponding to current INDEX of the activation table 1016 (step S1703). Here, at the time of processing initiation of CPU, only once, INDEX is initialized and it is referred to as INDEX=0. Then, since the event is not registered into a corresponding display event queue, it progresses to step S1708 and judges whether registration of the process of the task number corresponding to INDEX which can be performed is shown in the activation table 1016. Since there is also no registration of the process which can be performed, 1 is added to INDEX (step S1710), and it is referred to as INDEX=1. And INDEX judges whether it is more than the number of activation tables (= 3) (step S1711), since it still is not more than the number of activation tables, it progresses to step S1713, and the number of activation processes judges whether it is more than the number of the maximum activation (= 3) (step S1713). Since it still is not more than the number of the maximum activation, it progresses to step S1714 and loop count judges whether it is more than the number of activation tables. Since loop count still is not more than the number of activation tables, either, it adds return even to step S1702, adds 1 to loop count, and sets it to loop count =2. Since there is no event registered into the temperature monitor event queue of the task number corresponding to INDEX=1 this time also, it adds 1 to INDEX (step S1710), and only sets to INDEX=2, and returns to step S1702.
[0131] At the time of INDEX=2, since the queuing of the registration event "an air conditioning key" is carried out, in decision of step S1703, to the corresponding air conditioning event queue of a task number, it is judged as those with a registration event, and judges to it whether there is any task which is registered into the task activation table 1016 next and which can be performed (step S1704). Since nothing is registered into the activation table 1016, it progresses to step S1706 and the sequence activation section 1014 still gives [registering the head process corresponding to the head event of an event queue into the activation table 1016, and] an instruction to the process registration section 1015.
[0132] Since the air conditioning key is registered into the air conditioning event queue, the above-mentioned process registration section 1015 registers the compressor situation check which is the head process of this subsequence table into the process corresponding to the air conditioning task of the activation table 1016 which can be performed with reference to the air conditioning sequence of memory 1019 with reference to the compressor control subsequence table which is the head subsequence of an air conditioning sequence. At this time, the latency time and transition authorization which are indicated by coincidence at this subsequence table are also registered into the activation table 1016. The registered activation tables 1016, such as the above-mentioned process, are shown by drawing 15 (b). And the sequence activation section 1014 deletes an air conditioning key from an air conditioning event queue (step S1707). Next, it progresses to step S1708, and although it judges whether the process which can be performed is shown in the activation table 1016, since there is a compressor situation check registered previously, those with an activation process, a next door, and predefined process "a processes run" are started (step S1709).
[0133] The predefined process "a processes run" of step S1709 is shown by the flow chart of drawing 18 . First, the latency time of the task number corresponding to the present INDEX of the activation table 1016 in the sequence activation section 1014 = it judges whether it is 0 (step S1801), and when it is not latency-time =0, a processes run is ended, it progresses to step S1710 of the flow chart of drawing 17 R> 7, and 1 is added to INDEX. This time, since it is latency-time =0, a compressor situation check is taken out from the process of the

activation table 1016 which can be performed, this process is deleted from the above-mentioned activation table 1016 (step S1802), and this process is performed (step S1803). and the number of activation processes -- 1 -- adding -- the number of activation processes -- it is referred to as =1.

[0134] Next, the activation result of a compressor situation check is returned to the sequence activation section 1014. And it judges whether the sequence activation section 1014 has the normal activation result of the above-mentioned compressor situation check (step S1805), and when not normal, an instruction is given to the process registration section 1015 so that the process of a corresponding transition place may be registered into the activation table 1016 with reference to the transition table corresponding to a compressor situation check. The process registration section 1015 registers a transition place process into the activation table 1016 with this instruction (step S1808).

[0135] Occasionally, it judges [whose activation result is normal] whether the sequence activation section 1014 has a process [degree] process in the process registration section 1015 (step S1806), and in a certain case, an instruction is issued so that the process may be registered into the activation table 1016 (step S1807). Since there is engine-speed modification as a process [degree] process this time, the process registration section 1015 registers this process into the process of the activation table 1016 which can be performed. And the processes run of step S1709 is ended, 1 is added to INDEX, and it is referred to as INDEX=3 (step S1710). And it judges that INDEX is more than the number of activation tables in step S1711, and replaces with INDEX=0.

[0136] Next, it progresses to step S1714 and it is judged that loop count is more than the number of activation tables. In addition, also when, as for decision of step S1714, the process is not registered into the activation table 1016, the sequence activation section 1014 is decision required in order to end processing of the time, when seeing briefly the column of the process of the activation table 1016 which can be performed. If there is no decision of step S1714, when there will be no registration of a process in the activation table 1016, the sequence activation section 1014 checks the activation table 1016 eternally, and cannot end processing of the time. And it progresses to step S1715, only processing spacing time amount (= 5ms) lengthens the latency time of all the processes of the activation table 1016 registered, and this sequence executive operation is ended. Initiation of next sequence executive operation carries out sequential execution of the processing from step S1701.

[0137] Thus, according to the sequence control equipment by the gestalt 3 of this operation, and the approach By carrying out scheduling of the process corresponding to two or more tasks which should be performed with INDEX and a task number, and carrying out sequential execution of the data of a layered structure by the program execution section 1004 The process which remains being able to perform two or more sequence control in parallel, and not performing will not exist, but the size of a data program itself can be further managed with necessary minimum. Moreover, since the data of the gestalt 3 of this operation are only expressed by the table of a layered structure, easily, the addition of a process and deletion are performed and a change etc. can be made, and versatility is high and it becomes possible to realize the sequence control equipment which was rich in flexibility, and an approach. Furthermore, since the number of the maximum processes performed by one processing is controllable by step S1713 of drawing 1717 according to the capacity of CPU, the situation which activation of a process does not end within 1 time of the processing time is also avoidable.

[0138] Moreover, before performing sequence processing corresponding to the event even if other events occur by carrying out transition authorization of the process to prohibition when there is a process which must perform it in a sequence since transition authorization of a subsequence table can be performed with "prohibition" or "authorization", the above-mentioned process can be performed.

[0139] In addition, by the sequence control equipment of the gestalt 3 of this operation, and the approach, although only the sequence control corresponding to an event task was explained, this is an example, for example, sequence control corresponding to a resident task and a fixed-cycle task can also be performed, and the same effectiveness is acquired. As the approach of the sequence control of a resident task and a fixed-cycle task, it has the periodic event generator made to generate an event periodically, or the process [degree] process of the process of the last of a sequence is made into the head process of a sequence, and, in the case of a resident task, the latency time of this head process is set to 0, and, in the case of a fixed-cycle task, the latency time of this head process should just be determined in consideration of a period.

[0140] Moreover, although the number of activation tables shall be one and the process corresponding to all tasks shall be registered into the above-mentioned activation table with the gestalt 3 of this operation in registration of the process which can be performed It is also possible to set up the number of activation processes which this is an example, for example, divides all tasks into the group of two or more tasks, and is

equipped with an activation table for this every group, and is performed by one processing for every group. At this time, from initiation of the flow chart of drawing 17 R> 7 to termination will be repeated for every group of a task in one processing. Therefore, when the number of the groups of the above-mentioned task is three, it is necessary to prepare three kinds of INDEX(s) like INDEX (1), INDEX (2), and INDEX (3) for every group. In addition to the same effectiveness as the gestalt 3 of this operation, a sequence with the above-mentioned high priority has the effectiveness performed more certainly by assigning more numbers of activation processes to the group of a task including a sequence with a still higher priority by having an activation table for every specific group of a task as mentioned above, and setting up the number of activation processes for every above-mentioned group. Moreover, it is also possible to carry out the group division of the task by drive classification of a task, and there is effectiveness performed by being stabilized at this time, without a fixed-cycle task and a resident task being barred by the activation of an event task started irregularly.

[0141] Moreover, with the gestalt 3 of this operation, although sequence control was expressed with two hierarchies of a sequence table 1007 and the subsequence table 1008, this is an example, and it is only a sequence table according to the amount of the process included in sequence control, or it is possible to realize also by using three or more hierarchies' sequence table, a subsequence table, etc., and the same effectiveness is acquired.

[0142] Moreover, although [the gestalt 3 of this operation / data division 1001 the program generation section 1002, the storage section 1003, and the program execution section 1004] all are contained in one sequence control equipment 1000 This is an example and the program recorded on the program generation equipment 1900 which generates a program, the record medium 1903 which records the program by which generation was carried out [above-mentioned], and the above-mentioned record medium 1903 is read like drawing 19 . It is also possible to divide into the program execution equipment 1901 which performs sequence control. In addition, let the approach of performing sequence control for the approach of generating a program with the above-mentioned program generation equipment 1900 with a program generation method and the above-mentioned program execution equipment 1901 be the program execution approach. Here, in drawing 19 , the same sign as drawing 10 shows the same thing as the sequence control equipment of the gestalt 3 of this operation, and those explanation is omitted. The write-in section 1902 writes the program which the program generation section 1002 generated in a record medium 1903. An executable program is written in the above-mentioned record medium 1903. Moreover, the reading section 1904 reads the executable program written in the record medium 1903. The read program is stored in the memory which the above-mentioned reading section 1904 does not illustrate.

[0143] Since only program execution equipment 1901 can be further used independently on the same effectiveness as the gestalt 3 of this operation by dividing the sequence control equipment 1000 of the gestalt 3 of this operation into program generation equipment 1900 like drawing 19 , program execution equipment 1901, and a record medium 1903, the part and the equipment which do not contain program generation equipment 1900 compared with sequence control equipment 1000 can be miniaturized. Moreover, since it is also possible to realize sequence control with two or more program execution equipments 1901 even if there is only single program generation equipment 1900, the cost can be cut down compared with the case where two or more sequence control equipments 1000 are used. Furthermore, the sequence control of program execution equipment 1901 can also be easily changed only by modification of a record medium 1903 by recording the data program of the sequence control of two or more classes on the record medium 1903 with program generation equipment 1900 in advance.

[0144] Moreover, although the number of the events which can carry out a queuing to each event queue of the event queue table 1013 is written by drawing 13 (a) as three pieces with the gestalt 3 of this operation, if it is the number which can carry out the queuing only of the required event by the system, this will be an example, it is not restricted to three pieces, and you may be four pieces and five pieces or more, for example, the same effectiveness will be acquired.

[0145] Moreover, with the gestalt 3 of this operation, although the transition place process on the transition table 1009 shall be specified by assignment of a transition place sequence table, a subsequence number, and a process number, this is an example, for example, by specifying a subsequence table and a process, a transition place process can also be specified and the same effectiveness is acquired.

[0146] Moreover, although [the gestalt 3 of this operation / INDEX] a task number and INDEX are continuous integers which begin from 0, and INDEX is shifted when only 1 makes this INDEX increase The figure which changes with some fixed regulations, such as a continuous integer which this is an example, for example, begins

from 1, or continuous even number which begins from 0, and not overlapping, Or if it is a notation etc., it will be possible to also use them as a task number and INDEX and for it to be possible, and to decrease INDEX further, or to also make it change according to a certain fixed regulation, and the same effectiveness will be acquired.

[0147] Moreover, with the gestalt 3 of this operation, although processing which shifts one INDEX is performed after activation of a process, it is also possible to perform the processing which this is an example, for example, shifts one INDEX in drawing 17 , step S1711, and step S1712 in front of the activation S1708 of a process, i.e., a step, etc., and the same effectiveness is acquired.

[0148] Moreover, with the gestalt 3 of this operation, although processing spacing of a processes run was set to 5ms, this is an example, no matter it may be what processing spacing, control of the sequence included in two or more tasks can be realized, and equivalent effectiveness is acquired.

[0149] Moreover, although [the gestalt 3 of this operation] processes run processing is performed periodically, even if such, the scheduling of the task using a task number and INDEX is possible [this is an example, for example, it is also possible to perform processes run processing continuously, and].

[0150] Moreover, although [the gestalt 3 of this operation] a process is taken out from an activation table, is deleted and is performed in the flow chart of drawing 18 , this is an example, and after it performs a process, it may delete this process from the above-mentioned activation table. Or overwriting is also possible, in case the process to delete is skipped and a process [degree] process or a transition place process is registered into an activation table.

[0151] Moreover, with the gestalt 3 of this operation, a process [degree] process is an unit, although the transition place process should be chosen as the transition table based on the activation result from the multiple processes of a publication, this is an example, it is also possible to choose a process [degree] process when the activation result of a process is normal from multiple processes, and the same effectiveness is acquired.

[0152] In addition, also when the record medium which recorded the program which realizes the task-scheduling approach shown with the gestalt of each above-mentioned implementation, the sequence-control approach, a program generation method, and the program execution approach supplies to a system or equipment and the main processing sections, such as the CPU of the system or equipment, read and perform the program stored in this record medium, the effectiveness which explained with the gestalt of each above-mentioned implementation, and the same effectiveness can acquire.

[0153] In addition, as a record medium which records a program, a floppy disk, a hard disk, an optical disk, a magnetic disk, a magneto-optic disk, CD-ROM, a magnetic tape, a punch card, the memory card of a non-volatile, ROM, etc. can be used, for example.

[0154]

[Effect of the Invention] It is effective in the ability to prevent the situation that a specific task remains not performing possible [without performing fine control by the complicated program by having considered as the approach of carrying out scheduling of the task using a task number and INDEX according to the task-scheduling approach of claim 1 and claim 2 / carrying out scheduling of the task] so that more clearly than the above explanation.

[0155] Moreover, according to the task-scheduling approach of claim 3, in addition to the effectiveness of above-mentioned claims 1 and 2, further, activation of a task is not completed in 1 time of the processing time, but it is effective in the situation that the starting timing of a resident task and a fixed-cycle task is not securable being avoidable with the ability of the number of activation tasks in one executive operation to be restricted.

[0156] moreover -- according to the task-scheduling approach of claim 4 and claim 5 -- the drive classification of a task -- or By having considered as the approach that the number of activation tasks can be restricted for every group of a task It adds to the effectiveness of above-mentioned claim 3. Activation of a fixed-cycle task and a resident task A task with the above-mentioned high priority has the effectiveness which comes to be performed more certainly by being able to prevent being barred by the activation of an event task started irregularly, and assigning the group of a task with a high priority more numbers of activation tasks.

[0157] By moreover, the thing which shall be ended by checking the task corresponding to the total task number of a task activation table for activation of the task in one processing which can be performed according to the task-scheduling approach of claim 6 The situation which cannot end processing is avoidable until it can end one processing and then a task is registered into a task activation table, also when the task is not registered into a task activation table.

[0158] Moreover, according to the task-scheduling approach of claim 7, there is effectiveness which can carry

out scheduling by performing registration to the task activation table of a task according to generating of a task activate request, without generating the specific task which is not equally performed by the above-mentioned task-scheduling approach in the task which the activate request of a task generated.
[0159] Moreover, the task activation table on which the task which can be performed is registered for every task number according to the task-scheduling equipment of claim 8, The task registration section which registers into the above-mentioned task activation table the task which the task activate request generated as a task which can be performed, By having had the task activation section which performs the task of the task number corresponding to an index which can be performed, shifting an index Even if it does not perform fine control, the sequential execution of the task can be carried out using an index and a task number, and it is effective in the situation which is left with a specific task not performed being avoidable.
[0160] Moreover, according to the program generation method of claim 9 thru/or claim 12, the data with which sequence control was described are read. By generating an executable program by changing the contents of the above-mentioned data into a program source code, and compiling and linking the above-mentioned program source code The program created by this program generation method Since it consists of only data, such as a transition place from the treatment process contained in a sequence, and this process, the capacity of a program can be managed with the minimum. Moreover, the above-mentioned data Since the process of sequence control etc. is only described, it is effective in the ability to respond to modification of a sequence etc. easily.
[0161] Moreover, according to the program generation equipment of claim 13 thru/or claim 16 By having had the data division sequence control was described to be, the program generation section which generates a program based on the above-mentioned data division, and the write-in section which records the above-mentioned program on a record medium Since the program recorded on the above-mentioned record medium is only data, the capacity can be managed with the minimum, and since the process of sequence control etc. is only described, the above-mentioned data division are effective in the ability to respond to modification of a sequence etc. easily.
[0162] Moreover, according to the program execution approach of claim 17 thru/or claim 19 It has the sequence information two or more treatment processes contained in a sequence and the transition place from this treatment process were described to be. It is the program execution approach of performing sequence control, and sequential execution of the process of sequence control is carried out with reference to the above-mentioned sequence information. When the activation result of the above-mentioned process is not normal By performing a transition place process with reference to the above-mentioned sequence information, it is effective in the ability to perform complicated sequence processing using the sequence information only on data.
[0163] Moreover, according to the program execution approach of claim 20 thru/or claim 24, claim 28, and claim 30 A task number is given to a task. The relation between each task name and a sequence, The sequence information two or more treatment processes contained in this sequence and the transition place from this treatment process were described to be, And the process which can be performed has the activation table on which this process belongs and on which it is registered for every task. It is the program execution approach which carries out sequential execution of the process which was registered into the above-mentioned table which can be performed, and which can be performed while carrying out scheduling of the task. When performing one of sequence control The process included in this sequence control is registered into an activation table as a process which can be performed. The above-mentioned process which can be performed is performed in order of a task number, and degree treatment process is registered into the above-mentioned activation table with reference to the above-mentioned sequence information as a process which can be performed based on the activation result of the above-mentioned process. The above-mentioned process registration which can be performed, It is effective in the ability to perform two or more sequence processings in parallel, without producing the specific process which becomes [performing with as using the sequence information only on data, without controlling by the complicated program by repeating activation of this process and performing it, and].
[0164] Moreover, according to the program execution approach of claim 25 and claim 26, it sets to the program execution approach according to claim 24. a limit of the number of processes performed by one processing -- every drive classification of a task -- or By carrying out for every group of a task, activation of a fixed-cycle task and a resident task A sequence with the above-mentioned high priority has the effectiveness performed more certainly by assigning more numbers of activation processes to the group of a task including a sequence with a high priority, without being barred by the activation of an event task started irregularly.
[0165] Moreover, according to the program execution approach of claim 27, it sets to the program execution

approach of any one publication of claim 24 thru/or claim 26. In the above-mentioned processing, if the process of the total task number of the above-mentioned activation table which can be performed is checked, by ending processing of the time The situation which cannot end processing is avoidable until it can end one processing and then a process is registered into an activation table, also when the process is not registered into an activation table.

[0166] Moreover, according to the program execution approach of claim 29, in the program execution approach of any one publication of claim 17 thru/or claim 28, it is effective in the ability to change the contents of the control sequence easily only by changing the above-mentioned record medium by having made the above-mentioned sequence information into the executable program recorded on the record medium in addition to the effectiveness of above-mentioned claim 17 thru/or claim 28.

[0167] Moreover, according to the program execution equipment of claim 31 thru/or claim 33, it is effective in the ability to perform complicated sequence processing with the program of only data by having had the reading section which reads the program written in the record medium, and the program execution section which performs sequence control by the program read the account of a top.

[0168] Moreover, according to the program execution equipment of claim 34 thru/or claim 37 The relation of the each task name and sequence which were written in the record medium, and the relation between the above-mentioned sequence and an event, With the reading section which reads a program including the sequence information two or more treatment processes contained in the above-mentioned sequence and the transition place from this treatment process were described to be, the account of a top by the read program In program execution equipment equipped with the program execution section which performs sequence control the above-mentioned program execution section By performing scheduling of a task with activation of sequence control using a task number and an index It is effective in the ability to perform two or more sequence processings in parallel, without producing the specific process which becomes [performing with as, and] using the sequence information only on data.

[0169] Moreover, according to the sequence control approach of claim 38 thru/or claim 40 It is the sequence control approach of having data with which sequence control was described and performing this sequence control. By reading the contents of the above-mentioned data, changing into a program, storing in the storage section, and carrying out sequential execution of each process of sequence control with reference to the above-mentioned program Since the above-mentioned program is only data, the capacity can be managed with the minimum, and since the above-mentioned data are [that the process of sequence control etc. is only described, and], they are effective in performing the addition of a process and being able to make a change etc. easily. Furthermore, it is effective in the ability to perform complicated sequence control by referring to the program of only these data.

[0170] Moreover, according to the sequence control equipment of claim 41 thru/or claim 45 The data division which have data with which two or more treatment processes contained in a sequence and the transition place from this treatment process were described, The program generation section which reads the contents of the above-mentioned data division and is changed into a program, With reference to the above-mentioned program stored in the storage section in which the above-mentioned program is stored, and the above-mentioned storage section, sequential execution of each process of sequence control is carried out. By having had the program execution section which performs degree treatment process with reference to the above-mentioned program based on the activation result of the above-mentioned process, the above-mentioned program Since it is only data, the capacity can be managed with the minimum, and since the above-mentioned data are [that the process of sequence control etc. is only described, and], they are effective in performing the addition of a process and being able to make a change etc. easily. Furthermore, it is effective in the ability to perform complicated sequence control by referring to the program of only these data.

[0171] Moreover, according to the task-scheduling record medium of claim 46, it has the task activation table on which the task which can be performed is registered. By having recorded the program performed in order of the task number to which the above-mentioned task which can be performed was given by the task in the task-scheduling program documentation medium which recorded the program which controls the execution sequence of the above-mentioned task which can be performed There is effectiveness which can be offered by the small program of capacity without considering the task-scheduling approach which does not produce the specific task which is not performed as the program which performs fine control.

[0172] Moreover, two or more treatment processes which are contained in a sequence according to the program generator record medium of claim 47, Read the data with which the transition place from this treatment process

was described, and the contents of the above-mentioned data are changed into a program source code. By having recorded the program which generates an executable program by compiling and linking the above-mentioned program source code, the above-mentioned executable program Since it consists of only data, the programming approach which creates the executable program used in the sequence control of this invention by the minimum capacity can be supplied to a personal computer or a workstation.

[0173] Moreover, according to the program execution program documentation medium of claim 48 and claim 49 With reference to the sequence information about sequence control, carry out sequential execution of the process of sequence control, and based on the activation result of the above-mentioned process by having recorded the program which performs degree treatment process with reference to the above-mentioned sequence information Using the sequence information only on the above-mentioned data, the program execution approach that complicated sequence control can be performed can be supplied to a personal computer, a workstation, or a device inclusion mold microcomputer.

[0174] Moreover, two or more treatment processes which are contained in a sequence according to the sequence control program documentation medium of claim 50, It is the sequence control program documentation medium which recorded the program which has data with which the transition place from this treatment process was described, and performs the above-mentioned sequence control. Read the contents of the above-mentioned data, change into a program, and it stores in the storage section. With reference to the program of the above-mentioned storage section, carry out sequential execution of each process of sequence control, and based on the activation result of the above-mentioned process by having recorded the program which performs degree treatment process with reference to the above-mentioned program The program recorded on the above-mentioned storage section can be managed with the minimum capacity of only data. Moreover, the above-mentioned data By performing the addition of a process, being able to make a change etc. easily, and referring to the program of only these data further, since it is [that the process of sequence control etc. is only described, and] The sequence control approach that complicated sequence control can be performed can be supplied to a personal computer, a workstation, or a device embedded-type microcomputer.

[0175] Moreover, according to the record medium of claim 51 thru/or claim 53, it is effective in the ability to relate with the field which records a process at the above-mentioned process, and offer the record medium with which the data of sequence control used in the program execution approach of this invention by having had the unit or the field which records two or more transition place processes were recorded by the minimum capacity.

[Translation done.]

* NOTICES *

1.This document has been translated by computer. So the translation may not reflect the original precisely.
2.**** shows the word which can not be translated.
3.In the drawings, any words are not translated.

TECHNICAL FIELD

[Field of the Invention] This invention relates to the task-scheduling approach which does not need fine control especially, task-scheduling equipment, a program generation method, program generation equipment, the program execution approach, program execution equipment, the sequence control approach, sequence control equipment, and a record medium.

[Translation done.]

* NOTICES *

1.This document has been translated by computer. So the translation may not reflect the original precisely.
2.**** shows the word which can not be translated.
3.In the drawings, any words are not translated.

PRIOR ART

[Description of the Prior Art] In a microcomputer control program, the sequence control which carries out sequential execution of the operations sequence (sequence) decided beforehand is common, carrying out coordination actuation of the hardware of the connected circumference. In this sequence control, the state transition of being as redoing a certain processing **** [, and] occurs. [interrupting processing, if an error occurs for every processing] A sequence is assigned to the task which is the run unit of CPU, and it is made to operate in the program of a microcomputer, carrying out scheduling of two or more tasks. If a task is classified according to drive classification, it will become three kinds such as the resident task which is always operating, the fixed-cycle task started for every fixed time amount the period beforehand decided using timer ability etc., and the event task started by external events, such as a user input, from the time of a system startup.
[0003] First, the conventional task-scheduling approach is explained. As the conventional task-scheduling approach, round robin scheduling is common. This round robin scheduling is a method which assigns CPU equally to them, when it performs from what has an execution priority high when there are two or more tasks from which an execution priority differs and there are two or more tasks with an equal execution priority. that is, if CPU is assigned to a certain task and only the decided fixed time quantum passes when there are two or more tasks with an equal execution priority, the task will be terminated compulsorily -- having -- the following task with an equal execution priority -- receiving -- the above -- only a fixed time quantum is the method that CPU is assigned.
[0004] For example, CPU which is not illustrated performs processing every 5ms, and the relation between a task and its execution priority presupposes that it is given in the table of drawing 20 (a). When a task activate request occurs, each task is registered into a task activation table, and the registered task is performed by order with a high execution priority, i.e., descending of a value. Before CPU starts the 1st processing at 0ms, suppose that Task A - Task E were registered into the task activation table like drawing 20 (b) in this sequence. In this registration, a task is registered into a task activation table from top one as a task which can be performed. And a task is performed from a top thing and the performed task is deleted from a task activation table. When a top task is performed and deleted, the remaining tasks will shift to one top one. And if the high task of an execution priority which can be performed is lost, the task of the following execution priority which can be performed is performed. In addition, it is shown in drawing 20 (b) and (c) that "-" does not have the task which is registered and which can be performed. Here, since it is easy, it is [that generating of the task activate request of each task and registration to a task activation table are only performed once, and], and the task activate request of each task already presupposes that it did not generate more than this.
[0005] First, CPU performs the highest task A and Task B of an execution priority in the 1st processing for 0ms, and deletes them from a task activation table. Next, the task C with an execution priority high to the 2nd and Task D are performed, and they are also deleted from a task activation table. Next, the task E with an execution priority low No. 1 is performed, and this task E is deleted from a task activation table. And the 1st processing is ended. Next, since there is no task which was registered into the task activation table like drawing 20 (c) at the time of initiation of the 2nd processing 5ms after and which can be performed, only the check of a task activation table is performed.
[0006] Thus, a task can be performed in order by setting an execution priority as a task. By the task-scheduling approach by above-mentioned round robin scheduling, when there are two or more fixed-cycle tasks started periodically, or when a task is registered into a task activation table beyond the throughput of CPU, a specific task with a low execution priority may not be performed. In order to avoid such a situation, a designer needs to create the program for task scheduling by which each task is performed equally by setting up the starting period

and the execution priority or changing the execution priority of a task dynamically for every fixed period so that the starting timing and the execution priority of all tasks may be examined beforehand and there may be no task which is not performed.

[0007] Next, the program generation method of the conventional sequence control is explained. As a program generation method of sequence control, the approach of using a fixed sequence pattern, and the method of using a state transition table are learned. First, the program generation method using a fixed sequence pattern is explained. The program generation method using a fixed sequence pattern is the approach of fixing the whole structure as a sequence pattern, creating a sequence sequence table, an operating-condition table, etc. to the possible part of modification, making combine automatically these each table information and sequence patterns as the whole structure, and generating a sequence control program.

[0008] Hereafter, an example explains the program generation method using a fixed sequence pattern. First, the sequence sequence table of drawing 21 (a) and the operating-condition table of the processing B of drawing 21 R> 1 (b) are defined. Processing 1 - processing 3 are defined as Processing A - Processing C, conditions 1 - conditions 3, and a list, the contents of processing which interpret processing of a publication and the contents of a condition on this table, and correspond based on such table information, conditional branching, etc. include in the sequence pattern which it has beforehand, and the program generation method using a fixed sequence pattern generates a sequence-control program automatically as a whole in these tables.

[0009] Drawing 22 is a flow chart which shows an example of the sequence control program generated based on drawing 21 (a) and each table information on (b). Next, the sequence control program generated by the program generation method which uses a fixed sequence pattern using the flow chart of drawing 22 is explained. First, since the initiation latency time of Processing A is 0ms when processing is started, CPU performs processing A (step S2201). Next, waiting for time amount only of the 20ms only of the initiation latency times over Processing B is carried out (step S2202), they progress to step S2203, and it judges whether an operating condition fulfills conditions 1. Processing 1 will be performed if it is filling (step S2204). While not filling, or when processing 1 is performed, it judges whether an operating condition fulfills conditions 2 (step S2205). Processing 2 will be performed if it is filling (step S2206). While not filling, or when processing 2 is performed, it judges whether an operating condition fulfills conditions 3 (step S2207). Processing 3 will be performed if it is filling (step S2208). While not filling, or when processing 3 is performed, waiting for time amount only of the 5ms only of the initiation latency times over Processing C is carried out (step S2209), then they perform processing C (step S2210), and end all processings.

[0010] In addition, since actuation of equipment etc. is completed, or since equipment etc. is stabilized in the condition that the next processing can be performed, the initiation latency time over each processing in the above-mentioned explanation means fixed time amount defined to each processing, and waiting. Although it is also possible whether processing can be performed and or not for equipment to be supervised continuously, the configuration of equipment of the configuration of a program will also be simple, and the direction which sets up the initiation latency time will end. Thus, according to the program generation method using a fixed sequence pattern, the sequence control program shown by the flow chart of drawing 22 is automatically generable by indicating the desired contents of processing and the desired initiation latency time to drawing 21 (a), and indicating the contents of processing when fulfilling desired conditions and conditions on the operating-condition table of Processing B.

[0011] Next, the program generation method using a state transition table is explained. Drawing 23 is drawing showing the state transition table which expressed sequence control by the condition (S) and the matrix which consists of an event (E). First, the state transition table of drawing 23 defines a control sequence. In this table, the upper case of each cel shows the contents of processing, and below the arrow head of the lower berth shows the transition place of a condition (S). In addition, the contents of processing show that the cel of "-" does not process, and the cel of a null shows performing neither processing of what nor a state transition. It defines in which condition if which event happens in each condition, what kind of processing will be performed by the above-mentioned state transition table, and after the processing is completed, it changes. The approach of generating automatically those processings and the program which described transition of a condition using the above-mentioned state transition table is a program generation method using a state transition table.

[0012] The above-mentioned program generation method interprets the written contents of the table with reference to the state transition table of drawing 23 . A certain condition (S) when a certain processing or transition exists in a certain event (E) It is the approach of generating a sequence control program by specifying those conditions (S) and events (E) by conditional branching, and a program's describing that the processing

indicated and transition are performed, and performing it to the whole state transition table.

[0013] Drawing 24 is a flow chart which shows an example of the sequence control program generated based on the state transition table of drawing 23 . In addition, the contents of the control sequence are the same as explanation of the program generation method using a fixed sequence pattern. The sequence control program generated by the program generation method which uses a state transition table using the flow chart of drawing 24 is explained. First, if a program is started, it will carry out under a condition = halt (step S2401). Next, since it judged in **** during the condition = halt (step S2402) and carried out under the condition = halt at step S2401, it progresses to step S2403 and judges in event = sequence initiation and ****. Event = at the time of sequence initiation and **, processing A is performed and it considers as waiting for condition =20ms (step S2404). It is step S2402, and during a condition = halt, when it is judged that it does not come out, or when it is judged at step S2403 that an event is not sequence initiation, and when processing of step S2404 is performed, it judges in waiting and **** for condition =20ms (step S2405). condition = -- the waiting for 20ms -- it comes out, and if it is, it will judge in progress and **** for event =20ms (step S2406), and will consider as condition = processing A termination at the time of event =20ms progress and ** (step S2407). It is step S2405, and the waiting for condition =20ms, when it is judged that it does not come out, when not being judged as waiting for event =20ms at step S2406, or when step S2407 is performed, it judges in condition = processing A termination and **** (step S2408).

[0014] Condition = at the time of processing A termination and **, it judges in the event = conditions 1 and **** (step S2409). Event = processing 1 is performed at the time of conditions 1 and **, and it makes it waiting for condition =5ms (step S2410). Moreover, the event = conditions 1, when not coming out, or when processing of step S2410 is performed, it judges in the event = conditions 2 and **** (step S2411). Event = at the time of conditions 2 and **, processing 2 is performed and it considers as waiting for condition =5ms (step S2412). Event = conditions 2, when not coming out, or when processing of step S2412 is performed, it judges in the event = conditions 3 and **** (step S2413). Event = processing 3 is performed at the time of conditions 3 and **, and it makes it waiting for condition =5ms (step S2414).

[0015] step S2408 -- condition = -- when [the event = conditions 3 and when / when it processed A ends and it is judged that it does not come out, or / not coming out], or, when processing of step S2414 is performed, it judges in waiting and **** for condition =5ms (step S2415). It is an event when a condition is the waiting for 5ms. It judges in progress and **** for =5ms (step S2416), and considers as condition = processing B termination at the time of event =5ms progress and ** (step S2417). step S2415 -- setting -- the waiting for condition =5ms, and the time of not coming out -- step S2416 -- setting -- event = -- for 5ms, it passes, and when not coming out, or when step S2417 is performed, it judges in condition = processing B termination and **** (step S2418). Condition = it judges in event = processing B termination and **** at the time of processing B termination and ** (step S2419), and processing C is performed at the time of processing B termination, and it considers it as condition = processing C termination (step S2420). step S2418 -- setting -- condition = -- the time of not processing, B ending and coming out -- step S2419 -- setting -- event = -- it processes B ends, and when not coming out, or when step S2420 is performed, it judges in condition = processing C termination and **** (step S2421). Condition = a program is ended at the time of processing C termination and **. however -- otherwise, -- coming -- being alike -- the processing from return and step S2401 is again repeated to step S2401.

[0016] Thus, according to the program generation method using a state transition table, sequence control is expressed by the matrix which consists of the condition (S) and event (E) like drawing 23 . By generating a program like the flow chart of drawing 24 which specifies a condition (S) and an event (E) by conditional branching using the above-mentioned state transition table, and performs processing and transition When an addition, deletion, modification, etc. change an event (E) into a condition (S) in the above-mentioned state transition table, easily, the addition of the processing in sequence control and deletion are performed, and a change etc. can be made, and can offer the high program generation method of versatility and description capacity. In addition, the sequence control program generated by the above-mentioned program generation method is compiled and linked, and turns into a program which can be performed.

[Translation done.]

* NOTICES *

1.This document has been translated by computer. So the translation may not reflect the original precisely.
2.**** shows the word which can not be translated.
3.In the drawings, any words are not translated.

EFFECT OF THE INVENTION

[Effect of the Invention] It is effective in the ability to prevent the situation that a specific task remains not performing possible [without performing fine control by the complicated program by having considered as the approach of carrying out scheduling of the task using a task number and INDEX according to the task-scheduling approach of claim 1 and claim 2 / carrying out scheduling of the task] so that more clearly than the above explanation.
[0155] Moreover, according to the task-scheduling approach of claim 3, in addition to the effectiveness of above-mentioned claims 1 and 2, further, activation of a task is not completed in 1 time of the processing time, but it is effective in the situation that the starting timing of a resident task and a fixed-cycle task is not securable being avoidable with the ability of the number of activation tasks in one executive operation to be restricted.
[0156] moreover -- according to the task-scheduling approach of claim 4 and claim 5 -- the drive classification of a task -- or By having considered as the approach that the number of activation tasks can be restricted for every group of a task It adds to the effectiveness of above-mentioned claim 3. Activation of a fixed-cycle task and a resident task A task with the above-mentioned high priority has the effectiveness which comes to be performed more certainly by being able to prevent being barred by the activation of an event task started irregularly, and assigning the group of a task with a high priority more numbers of activation tasks.
[0157] By moreover, the thing which shall be ended by checking the task corresponding to the total task number of a task activation table for activation of the task in one processing which can be performed according to the task-scheduling approach of claim 6 The situation which cannot end processing is avoidable until it can end one processing and then a task is registered into a task activation table, also when the task is not registered into a task activation table.
[0158] Moreover, according to the task-scheduling approach of claim 7, there is effectiveness which can carry out scheduling by performing registration to the task activation table of a task according to generating of a task activate request, without generating the specific task which is not equally performed by the above-mentioned task-scheduling approach in the task which the activate request of a task generated.
[0159] Moreover, the task activation table on which the task which can be performed is registered for every task number according to the task-scheduling equipment of claim 8, The task registration section which registers into the above-mentioned task activation table the task which the task activate request generated as a task which can be performed, By having had the task activation section which performs the task of the task number corresponding to an index which can be performed, shifting an index Even if it does not perform fine control, the sequential execution of the task can be carried out using an index and a task number, and it is effective in the situation which is left with a specific task not performed being avoidable.
[0160] Moreover, according to the program generation method of claim 9 thru/or claim 12, the data with which sequence control was described are read. By generating an executable program by changing the contents of the above-mentioned data into a program source code, and compiling and linking the above-mentioned program source code The program created by this program generation method Since it consists of only data, such as a transition place from the treatment process contained in a sequence, and this process, the capacity of a program can be managed with the minimum. Moreover, the above-mentioned data Since the process of sequence control etc. is only described, it is effective in the ability to respond to modification of a sequence etc. easily.
[0161] Moreover, according to the program generation equipment of claim 13 thru/or claim 16 By having had the data division sequence control was described to be, the program generation section which generates a program based on the above-mentioned data division, and the write-in section which records the above-mentioned program on a record medium Since the program recorded on the above-mentioned record medium is

only data, the capacity can be managed with the minimum, and since the process of sequence control etc. is only described, the above-mentioned data division are effective in the ability to respond to modification of a sequence etc. easily.

[0162] Moreover, according to the program execution approach of claim 17 thru/or claim 19 It has the sequence information two or more treatment processes contained in a sequence and the transition place from this treatment process were described to be. It is the program execution approach of performing sequence control, and sequential execution of the process of sequence control is carried out with reference to the above-mentioned sequence information. When the activation result of the above-mentioned process is not normal By performing a transition place process with reference to the above-mentioned sequence information, it is effective in the ability to perform complicated sequence processing using the sequence information only on data.

[0163] Moreover, according to the program execution approach of claim 20 thru/or claim 24, claim 28, and claim 30 A task number is given to a task. The relation between each task name and a sequence, The sequence information two or more treatment processes contained in this sequence and the transition place from this treatment process were described to be, And the process which can be performed has the activation table on which this process belongs and on which it is registered for every task. It is the program execution approach which carries out sequential execution of the process which was registered into the above-mentioned table which can be performed, and which can be performed while carrying out scheduling of the task. When performing one of sequence control The process included in this sequence control is registered into an activation table as a process which can be performed. The above-mentioned process which can be performed is performed in order of a task number, and degree treatment process is registered into the above-mentioned activation table with reference to the above-mentioned sequence information as a process which can be performed based on the activation result of the above-mentioned process. The above-mentioned process registration which can be performed, It is effective in the ability to perform two or more sequence processings in parallel, without producing the specific process which becomes [performing with as using the sequence information only on data, without controlling by the complicated program by repeating activation of this process and performing it, and].

[0164] Moreover, according to the program execution approach of claim 25 and claim 26, it sets to the program execution approach according to claim 24. a limit of the number of processes performed by one processing -- every drive classification of a task -- or By carrying out for every group of a task, activation of a fixed-cycle task and a resident task A sequence with the above-mentioned high priority has the effectiveness performed more certainly by assigning more numbers of activation processes to the group of a task including a sequence with a high priority, without being barred by the activation of an event task started irregularly.

[0165] Moreover, according to the program execution approach of claim 27, it sets to the program execution approach of any one publication of claim 24 thru/or claim 26. In the above-mentioned processing, if the process of the total task number of the above-mentioned activation table which can be performed is checked, by ending processing of the time The situation which cannot end processing is avoidable until it can end one processing and then a process is registered into an activation table, also when the process is not registered into an activation table.

[0166] Moreover, according to the program execution approach of claim 29, in the program execution approach of any one publication of claim 17 thru/or claim 28, it is effective in the ability to change the contents of the control sequence easily only by changing the above-mentioned record medium by having made the above-mentioned sequence information into the executable program recorded on the record medium in addition to the effectiveness of above-mentioned claim 17 thru/or claim 28.

[0167] Moreover, according to the program execution equipment of claim 31 thru/or claim 33, it is effective in the ability to perform complicated sequence processing with the program of only data by having had the reading section which reads the program written in the record medium, and the program execution section which performs sequence control by the program read the account of a top.

[0168] Moreover, according to the program execution equipment of claim 34 thru/or claim 37 The relation of the each task name and sequence which were written in the record medium, and the relation between the above-mentioned sequence and an event, With the reading section which reads a program including the sequence information two or more treatment processes contained in the above-mentioned sequence and the transition place from this treatment process were described to be, the account of a top by the read program In program execution equipment equipped with the program execution section which performs sequence control the above-mentioned program execution section By performing scheduling of a task with activation of sequence control

using a task number and an index It is effective in the ability to perform two or more sequence processings in parallel, without producing the specific process which becomes [performing with as, and] using the sequence information only on data.

[0169] Moreover, according to the sequence control approach of claim 38 thru/or claim 40 It is the sequence control approach of having data with which sequence control was described and performing this sequence control. By reading the contents of the above-mentioned data, changing into a program, storing in the storage section, and carrying out sequential execution of each process of sequence control with reference to the above-mentioned program Since the above-mentioned program is only data, the capacity can be managed with the minimum, and since the above-mentioned data are [that the process of sequence control etc. is only described, and], they are effective in performing the addition of a process and being able to make a change etc. easily. Furthermore, it is effective in the ability to perform complicated sequence control by referring to the program of only these data.

[0170] Moreover, according to the sequence control equipment of claim 41 thru/or claim 45 The data division which have data with which two or more treatment processes contained in a sequence and the transition place from this treatment process were described, The program generation section which reads the contents of the above-mentioned data division and is changed into a program, With reference to the above-mentioned program stored in the storage section in which the above-mentioned program is stored, and the above-mentioned storage section, sequential execution of each process of sequence control is carried out. By having had the program execution section which performs degree treatment process with reference to the above-mentioned program based on the activation result of the above-mentioned process, the above-mentioned program Since it is only data, the capacity can be managed with the minimum, and since the above-mentioned data are [that the process of sequence control etc. is only described, and], they are effective in performing the addition of a process and being able to make a change etc. easily. Furthermore, it is effective in the ability to perform complicated sequence control by referring to the program of only these data.

[0171] Moreover, according to the task-scheduling record medium of claim 46, it has the task activation table on which the task which can be performed is registered. By having recorded the program performed in order of the task number to which the above-mentioned task which can be performed was given by the task in the task-scheduling program documentation medium which recorded the program which controls the execution sequence of the above-mentioned task which can be performed There is effectiveness which can be offered by the small program of capacity without considering the task-scheduling approach which does not produce the specific task which is not performed as the program which performs fine control.

[0172] Moreover, two or more treatment processes which are contained in a sequence according to the program generator record medium of claim 47, Read the data with which the transition place from this treatment process was described, and the contents of the above-mentioned data are changed into a program source code. By having recorded the program which generates an executable program by compiling and linking the above-mentioned program source code, the above-mentioned executable program Since it consists of only data, the programming approach which creates the executable program used in the sequence control of this invention by the minimum capacity can be supplied to a personal computer or a workstation.

[0173] Moreover, according to the program execution program documentation medium of claim 48 and claim 49 With reference to the sequence information about sequence control, carry out sequential execution of the process of sequence control, and based on the activation result of the above-mentioned process by having recorded the program which performs degree treatment process with reference to the above-mentioned sequence information Using the sequence information only on the above-mentioned data, the program execution approach that complicated sequence control can be performed can be supplied to a personal computer, a workstation, or a device inclusion mold microcomputer.

[0174] Moreover, two or more treatment processes which are contained in a sequence according to the sequence control program documentation medium of claim 50, It is the sequence control program documentation medium which recorded the program which has data with which the transition place from this treatment process was described, and performs the above-mentioned sequence control. Read the contents of the above-mentioned data, change into a program, and it stores in the storage section. With reference to the program of the above-mentioned storage section, carry out sequential execution of each process of sequence control, and based on the activation result of the above-mentioned process by having recorded the program which performs degree treatment process with reference to the above-mentioned program The program recorded on the above-mentioned storage section can be managed with the minimum capacity of only data. Moreover, the above-

mentioned data By performing the addition of a process, being able to make a change etc. easily, and referring to the program of only these data further, since it is [that the process of sequence control etc. is only described, and] The sequence control approach that complicated sequence control can be performed can be supplied to a personal computer, a workstation, or a device embedded-type microcomputer.

[0175] Moreover, according to the record medium of claim 51 thru/or claim 53, it is effective in the ability to relate with the field which records a process at the above-mentioned process, and offer the record medium with which the data of sequence control used in the program execution approach of this invention by having had the unit or the field which records two or more transition place processes were recorded by the minimum capacity.

[Translation done.]

* NOTICES *

1.This document has been translated by computer. So the translation may not reflect the original precisely.
2.**** shows the word which can not be translated.
3.In the drawings, any words are not translated.

TECHNICAL PROBLEM

[Problem(s) to be Solved by the Invention] As mentioned above, in order that the round robin scheduling which is the conventional task-scheduling approach might carry out scheduling of the task with an execution priority, when many task activate requests occurred at once, only the task with a high execution priority was performed by processing at each time, and the task with a low execution priority had the problem that it did not perform. For example, if the event task of priority is higher than a fixed-cycle task when the activate request of two or more fixed-cycle tasks started periodically and two or more event tasks occurs in coincidence, the case where a fixed-cycle task cannot operate an assignment period will occur. In order to avoid this, a designer needs to constitute a program or needs to change the execution priority of a task dynamically every activation of a task, and a fixed period so that the execution priority, the starting timing, and the starting period of all tasks may be examined beforehand and there may be no task which is not performed, and it is necessary to constitute a program so that a task may be performed equally. However, by these solution approaches, there was a problem that the burden of the designer of a task-scheduling program increased, and a program will become complicated, and the capacity of a program became large as the result.
[0018] Furthermore, by the conventional task-scheduling approach, in order to perform in order the task which was registered into the task activation table and which can be performed, for example, even if it performed processing every 5ms, when required for activation of all tasks 5ms or more, about a resident task or a fixed-cycle task, there was also a problem that the starting timing of the following task was not secured.
[0019] Moreover, the program generation method using a fixed sequence pattern can respond only to the sequence pattern decided beforehand. That is, in explanation in the conventional example, although Processing A can be changed into processing A', or it newly adds Processing D, it cannot change so that Processing C may be divided into conditions 4 - conditions 6 and processing 4 - processing 6 may be performed. Thus, the program generation method using a fixed sequence pattern had the narrow adaptation range, and since it needed to change a sequence pattern about the addition of processing etc., it had the problem that the program generation method itself had to be changed.
[0020] Moreover, although the program generation method using a state transition table had high description capacity as mentioned above and it could respond to the addition of processing, modification, etc. flexibly, the generated program had the problem that became the thing of the redundant expression with much conditional branching, and the capacity of a program became large. Furthermore, since the state transition table created since a condition (S) and one event (E) increase became large and increased conditional branching etc. by the addition of time amount conditions, such as an addition of processing, or the latency time, there was also a problem that the capacity of a program will increase. For example, when flow chart drawing 22 and drawing 24 expressing the same sequence control of a program are compared, it turns out that the program created using the state transition table is more complicated than the program created using the fixed sequence pattern. Moreover, since the activity which creates a state transition table turned into an activity which creates a program with an equal mostly after all when there is much conditional branching etc., there was also a problem that the advantage as a program generation method was lost.
[0021] It aims at offering the task-scheduling approach by which it is made in order that this invention may solve the above-mentioned trouble, and the situation that a specific task is not performed can be canceled, and the starting timing of a resident task and a fixed-cycle task is secured, and equipment. Moreover, this invention aims at offering the task-scheduling program documentation medium which recorded the program corresponding to the above-mentioned task-scheduling approach. Moreover, the capacity of a data program is small and this invention aims at offering the sequence control approach which can respond to the addition of a

process, modification, etc. easily, and equipment. Moreover, this invention aims at offering the sequence control program documentation medium which recorded the program corresponding to the above-mentioned sequence control approach. Moreover, this invention aims at offering the program generation method which can generate the program as data with a small capacity, and equipment. Moreover, this invention aims at offering the program generator record medium which recorded the program corresponding to the above-mentioned program generation method. Moreover, this invention aims at offering the program execution approach that a data program can be performed as sequence control, and equipment. Moreover, this invention aims at offering the program execution program documentation medium which recorded the program corresponding to the above-mentioned program execution approach. Moreover, this invention aims at offering the record medium which recorded the data which have the layered structure used in sequence control.

[Translation done.]

* NOTICES *

1.This document has been translated by computer. So the translation may not reflect the original precisely.
2.**** shows the word which can not be translated.
3.In the drawings, any words are not translated.

MEANS

[Means for Solving the Problem] In order to attain the above-mentioned purpose, the task-scheduling approach concerning claim 1 has the task activation table on which the task which can be performed is registered, and performs it in order of the task number to which the above-mentioned task which can be performed was given by the task in the task-scheduling approach which controls the execution sequence of the above-mentioned task which can be performed.
[0023] Moreover, the task-scheduling approach concerning claim 2 Give a task number to a task and it has the task activation table on which the task which can be performed is registered. It is the task-scheduling approach which controls the execution sequence of the above-mentioned task which can be performed. When the task of the task number corresponding to a current index which can be performed is registered into the above-mentioned task activation table, while performing this task It has the 1st step deleted from the above-mentioned task activation table, and the 2nd step which shifts the above-mentioned index, and the above 1st and the 2nd step are repeated.
[0024] Moreover, in the task-scheduling approach according to claim 1 or 2, the task-scheduling approach concerning claim 3 restricts the number of tasks performed by one processing, and performs the above-mentioned processing periodically.
[0025] Moreover, the task-scheduling approach concerning claim 4 restricts the number of tasks performed by one processing for every group of a task in the task-scheduling approach according to claim 3.
[0026] Moreover, the task-scheduling approach concerning claim 5 restricts the number of tasks performed by one processing for every drive classification of a task in the task-scheduling approach according to claim 3.
[0027] Moreover, in the task-scheduling approach of any one publication of claim 3 thru/or claim 5, the task-scheduling approach concerning claim 6 is one above-mentioned processing, and when the task of the total task number of a task activation table which can be performed is checked, it ends processing of the time.
[0028] Moreover, the task-scheduling approach concerning claim 7 performs registration of the task to the above-mentioned task activation table which can be performed to generating of a task activate request in the task-scheduling approach of any one publication of claim 1 thru/or claim 6.
[0029] Moreover, the task-scheduling equipment concerning claim 8 The task activation table on which it is task-scheduling equipment which gives a task number to a task and controls the execution sequence of a task using this task number and an index, and the task which can be performed is registered for every task number, It has the task registration section which registers into the above-mentioned task activation table the task which the task activate request generated as a task which can be performed, and the task activation section which performs the task of the task number corresponding to an index which can be performed while shifting an index.

[0030] Moreover, the program generation method concerning claim 9 reads the data with which two or more treatment processes which can be set to sequence control, and the transition place from this treatment process were described, changes the contents of the above-mentioned data into a program source code, compiles and links the above-mentioned program source code, and generates an executable program.
[0031] Moreover, in the program generation method according to claim 9, as for the program generation method concerning claim 10, the relation between a task name and the above-mentioned sequence is further described by the above-mentioned data.
[0032] Moreover, in the program generation method according to claim 9, as for the program generation method concerning claim 11, the relation between a task name and an event, the relation between this event and the above-mentioned sequence, and ** are further described by the above-mentioned data.

[0033] Moreover, in the program generation method of any one publication of claim 9 thru/or claim 11, the above-mentioned data are defined for the program generation method concerning claim 12 by the table of a layered structure.
[0034] Moreover, the program generation equipment concerning claim 13 is equipped with the data division two or more treatment processes contained in a sequence and the transition place from this treatment process were described to be, the program generation section which generates a program with reference to the above-mentioned data division, and the write-in section which records the above-mentioned program on a record medium.
[0035] Moreover, in program generation equipment according to claim 13, as for the program generation equipment concerning claim 14, the relation between a task name and the above-mentioned sequence is further described by the above-mentioned data division.
[0036] Moreover, in program generation equipment according to claim 13, as for the program generation equipment concerning claim 15, the relation between a task name and an event, the relation between this event and the above-mentioned sequence, and ** are further described by the above-mentioned data division.
[0037] Moreover, in the program generation equipment of any one publication of claim 13 thru/or claim 15, the above-mentioned data division are defined for the program generation equipment concerning claim 16 by the table of a layered structure.
[0038] Moreover, the program execution approach concerning claim 17 has the sequence information two or more treatment processes contained in a sequence and the transition place from this treatment process were described to be, is the program execution approach of performing sequence control, carries out sequential execution of the process of sequence control with reference to the above-mentioned sequence information, and performs degree treatment process with reference to the above-mentioned sequence information based on the activation result of the above-mentioned process.
[0039] Moreover, the program execution approach concerning claim 18 The sequence information the transition place from two or more treatment processes and these treatment processes which are contained in a sequence was described to be, It is the program execution approach which carries out sequential execution of the process which has the activation table which registers the process which can be performed and was registered into the above-mentioned activation table, and which can be performed. While performing the 1st step which registers the head process of sequence control into the above-mentioned activation table with reference to the above-mentioned sequence information as a process which can be performed, and the above-mentioned process which can be performed It is based on the 2nd step deleted from the above-mentioned activation table, and the activation result of the above-mentioned process. The above 2nd and the 3rd step are repeated until it has the 3rd step which registers degree treatment process of the above-mentioned sequence information into an activation table as a process which can be performed and degree treatment process of the above-mentioned sequence information is lost.
[0040] Moreover, it has the latency time on the program execution approach according to claim 18 and corresponding to [approach / concerning claim 19 / program execution] a process in the above-mentioned sequence information, and it is the 2nd step of the above and only the above-mentioned latency time carries out waiting for time amount before activation of the above-mentioned process.
[0041] Moreover, the program execution approach concerning claim 20 The sequence information with which gave the task number to the task and the transition place from the relation between each task name and a sequence, two or more treatment processes which can be set to this sequence, and this treatment process was described to be, The activation table on which the process which can be performed is registered for every task to which this process belongs, It is the program execution approach which carries out sequential execution of the process which ****(ed) and was registered into the above-mentioned table which can be performed, and which can be performed while carrying out scheduling of the task. When performing one of sequence control The process included in this sequence control is registered into an activation table as a process which can be performed. The above-mentioned process which can be performed is performed in order of a task number, and degree treatment process is registered into the above-mentioned activation table with reference to the above-mentioned sequence information as a process which can be performed based on the activation result of the above-mentioned process. Registration of the above-mentioned process which can be performed, Activation of this process is repeated and is performed.
[0042] Moreover, the program execution approach concerning claim 21 The sequence information with which gave the task number to the task and the transition place from the relation between each task name and a

sequence, the relation between the above-mentioned sequence and an event, two or more treatment processes contained in the above-mentioned sequence, and this treatment process was described to be, The activation table on which the process which can be performed is registered for every task to which this process belongs, It has an event queue for every task number. A task A task number, It is the program execution approach which carries out sequential execution of the process which was registered into the above-mentioned activation table while carrying out scheduling using the index, and which can be performed. When a registration event is in the event queue of the same task number as a current index and there is no process of this task number which can be performed in the above-mentioned activation table The 1st step which deletes the above-mentioned registration event and registers the head process of the sequence corresponding to this event into the above-mentioned activation table with reference to the above-mentioned sequence information as a process of the above-mentioned task number which can be performed, When the process of the same task number as a current index which can be performed is registered into the above-mentioned activation table The 2nd step deleted from the above-mentioned activation table while performing this process, When there is degree treatment process with reference to the above-mentioned sequence information based on the activation result of the above-mentioned process It has the 3rd step which registers this following treatment process into the above-mentioned activation table as a process which can be performed, and the 4th step which shifts an index, and the above 1st thru/or the 4th step are repeated.

[0043] Moreover, the program execution approach concerning claim 22 In the program execution approach according to claim 21 at the 1st step of the above When a registration event is in the event queue of the same task number as a current index and the process of this task number which can be performed is shown in the above-mentioned activation table It can set up whether it replaces with this process that can be performed and the head process of the sequence corresponding to the head event of the above-mentioned event queue is registered into the above-mentioned activation table by transition authorization of the above-mentioned process which can be performed.

[0044] Moreover, the program execution approach concerning claim 23 In the program execution approach according to claim 21 or 22 When registering into the above-mentioned activation table the process which can be performed, the latency time corresponding to this process is also registered. At the time of activation of this process, when the above-mentioned latency time is not 0, this process is not performed, but it has the step which updates the latency time of the above-mentioned activation table after the 4th step of the above.

[0045] Moreover, in the program execution approach of any one publication of claim 20 thru/or claim 23, the program execution approach concerning claim 24 restricts the number of processes performed by one processing, and performs the above-mentioned processing periodically.

[0046] Moreover, the program execution approach concerning claim 25 restricts the number of processes performed by one processing for every group of a task in the program execution approach according to claim 24.

[0047] Moreover, the program execution approach concerning claim 26 restricts the number of processes performed by one processing for every drive classification of a task in the program execution approach according to claim 24.

[0048] Moreover, in the program execution approach of any one publication of claim 24 thru/or claim 26, in the above-mentioned processing, the program execution approach concerning claim 27 ends processing of the time, when the process of the total task number of the above-mentioned activation table which can be performed is checked.

[0049] Moreover, in the program execution approach of any one publication of claim 17 thru/or claim 27, the above-mentioned sequence information of the program execution approach concerning claim 28 is an executable program.

[0050] Moreover, the program execution approach concerning claim 29 is the executable program by which the above-mentioned sequence information was recorded on the record medium in the program execution approach of any one publication of claim 17 thru/or claim 28.

[0051] Moreover, in the program execution approach of any one publication of claim 17 thru/or claim 29, the above-mentioned sequence information is defined for the program execution approach concerning claim 30 by the table of a layered structure.

[0052] Moreover, the program execution equipment concerning claim 31 is equipped with the program execution section which performs sequence control by the program read the account of a top with the reading section which reads the program written in the record medium.

[0053] Moreover, the program execution equipment concerning claim 32 includes the sequence information two or more treatment processes by which the above-mentioned program is included in a sequence, and the transition place from this treatment process were described to be in program execution equipment according to claim 31.

[0054] Moreover, the program execution equipment concerning claim 33 In program execution equipment according to claim 32 the above-mentioned program execution section The activation table on which the process which can be performed is registered, and the process registration section which registers into the above-mentioned activation table the process which can be performed with reference to the above-mentioned program, The above-mentioned process which can be performed is performed, and it judges whether the activation result is normal, and has the processes run section which makes a process [degree] process or a transition place process register into the above-mentioned process registration section as a process which can be performed.

[0055] Moreover, as for the program execution equipment concerning claim 34, two or more treatment processes by which the above-mentioned program is included in the relation between each task name and a sequence, the relation between the above-mentioned sequence and an event, and the above-mentioned sequence in program execution equipment according to claim 31, and the transition place from this treatment process include the described sequence information.

[0056] Moreover, in program execution equipment according to claim 34, as for the program execution equipment concerning claim 35, the above-mentioned program execution section also performs scheduling of a task with activation of sequence control using a task number and an index.

[0057] Moreover, the program execution equipment concerning claim 36 In program execution equipment according to claim 34 or 35 the above-mentioned program execution section The activation table on which the task number was given for the process which can be performed and which is registered for every task, The process registration section which registers into the above-mentioned activation table the process which can be performed with reference to the above-mentioned program, The event queue table on which the generated event has the event queue with which this event belongs, and which is registered for every task, The event analysis section which will register this event into the event queue corresponding to the task to which this event belongs with reference to the above-mentioned program if an event occurs, An instruction is given to the process registration section so that it may register with the above-mentioned activation table by making each process of the sequence control corresponding to a registration event into the process which can be performed. It has the sequence activation section which performs the process of the same task number as this index which can be performed, shifting an index.

[0058] Moreover, as for the above-mentioned program, the program execution equipment concerning claim 37 is making the layered structure in the program execution equipment of any one publication of claim 31 thru/or claim 36.

[0059] Moreover, the sequence control approach concerning claim 38 It has data with which two or more treatment processes contained in a sequence and the transition place from this treatment process were described. Are the sequence control approach of performing the above-mentioned sequence control, and the contents of the above-mentioned data are read. It changes into a program, and stores in the storage section, sequential execution of each process of sequence control is carried out with reference to the program of the above-mentioned storage section, and degree treatment process is performed with reference to the above-mentioned program based on the activation result of the above-mentioned process.

[0060] Moreover, the sequence control approach concerning claim 39 The data with which the task number was given to the task and the transition place from the relation between each task name and a sequence, the relation between the above-mentioned sequence and an event, two or more treatment processes contained in the above-mentioned sequence, and this treatment process was described, The activation table on which the process which can be performed is registered for every task to which this process belongs, It has an event queue for every task number. A task A task number, The 1st step which is the sequence control approach which carries out sequential execution of the above-mentioned activation ****, carrying out scheduling using an index, reads the contents of the above-mentioned data, changes into a program, and is stored in the storage section, When a registration event is in the event queue of the same task number as a current index and there is no process of this task number which can be performed in the above-mentioned activation table The 2nd step which deletes the above-mentioned registration event and registers the head process of the sequence corresponding to this event into the above-mentioned activation table with reference to the above-mentioned sequence information as a

process of the above-mentioned task number which can be performed, When the process of the same task number as a current index which can be performed is registered into the above-mentioned activation table The 3rd step deleted from the above-mentioned activation table while performing this process, When there is degree treatment process with reference to the above-mentioned sequence information based on the activation result of the above-mentioned process It has the 4th step which registers this following treatment process into the above-mentioned activation table as a process which can be performed, and the 5th step which shifts an index, and the above 2nd thru/or the 5th step are repeated.

[0061] Moreover, in the sequence control approach according to claim 38 or 39, the above-mentioned data are defined for the sequence control approach concerning claim 40 by the table of a layered structure.

[0062] Moreover, the sequence control equipment concerning claim 41 The data division two or more treatment processes contained in a sequence and the transition place from this treatment process were described to be, The program generation section which reads the contents of the above-mentioned data division and is changed into a program, With reference to the above-mentioned program stored in the storage section in which the above-mentioned program is stored, and the above-mentioned storage section, sequential execution of each process of sequence control is carried out. Based on the activation result of the above-mentioned process, it has the program execution section which performs degree treatment process with reference to the above-mentioned program.

[0063] Moreover, the sequence control equipment concerning claim 42 In sequence control equipment according to claim 41 the above-mentioned program execution section The activation table on which the process which can be performed is registered, and the process registration section which registers into the above-mentioned activation table the process which can be performed with reference to the program of the above-mentioned storage section, The above-mentioned process which can be performed is performed, and the activation result judges whether it is normal, and is equipped with the processes run section which makes a process [degree] process or a transition place process register into the above-mentioned process registration section as a process which can be performed.

[0064] Moreover, the sequence control equipment concerning claim 43 The relation between each task name and a sequence, and the relation between the above-mentioned sequence and an event, The data division two or more treatment processes contained in the above-mentioned sequence and the transition place from this treatment process were described to be, The contents of the above-mentioned data division are read and the above-mentioned program stored in the program generation section changed into a program, the storage section in which the above-mentioned program is stored, and the above-mentioned storage section is referred to. With activation of sequence control It has the program execution section which also performs scheduling of a task using a task number and an index.

[0065] Moreover, the sequence control equipment concerning claim 44 In sequence control equipment according to claim 43 the above-mentioned program execution section The activation table on which the task number was given for the process which can be performed and which is registered for every task, The process registration section which registers into the above-mentioned activation table the process which can be performed with reference to the above-mentioned program, The event queue table on which the generated event has the event queue with which this event belongs, and which is registered for every task, The event analysis section which will register this event into the event queue corresponding to the task to which this event belongs with reference to the above-mentioned program if an event occurs, An instruction is given to the process registration section so that it may register with the above-mentioned activation table by making each process of the sequence control corresponding to a registration event into the process which can be performed. It has the sequence activation section which performs the process of the same task number as this index which can be performed, shifting an index.

[0066] Moreover, in the sequence control equipment of any one publication of claim 41 thru/or claim 44, the above-mentioned data division are defined for the sequence control equipment concerning claim 45 by the table of a layered structure.

[0067] Moreover, the task-scheduling program documentation medium concerning claim 46 has the task activation table on which the task which can be performed is registered, and records the program performed in order of the task number to which the above-mentioned task which can be performed was given by the task in the task-scheduling program documentation medium which recorded the program which controls the execution sequence of the above-mentioned task which can be performed.

[0068] Moreover, the program generator record medium concerning claim 47 records the program which reads

the data with which two or more treatment processes contained in a sequence and the transition place from this treatment process were described, the program which changes the contents of the above-mentioned data into a program source code, and the program which compiles and links the above-mentioned program source code, and generates an executable program.

[0069] Moreover, the program execution program documentation medium concerning claim 48 It has the sequence information two or more treatment processes contained in a sequence and the transition place from this treatment process were described to be. The program which is the program execution program documentation medium which recorded the program which performs sequence control, and carries out sequential execution of the process of sequence control with reference to the above-mentioned sequence information, Based on the activation result of the above-mentioned process, the program which performs degree treatment process with reference to the above-mentioned sequence information is recorded.

[0070] Moreover, the program execution program documentation medium concerning claim 49 The sequence information with which gave the task number to the task and the transition place from the relation between each task name and a sequence, the relation between the above-mentioned sequence and an event, two or more treatment processes contained in the above-mentioned sequence, and this treatment process was described to be, The activation table on which the process which can be performed is registered for every task to which this process belongs, It has an event queue for every task number. A task A task number, It is the program execution program documentation medium which recorded the program which carries out sequential execution of the above-mentioned process which can be performed while carrying out scheduling using an index. When a registration event is in the event queue of the same task number as a current index and there is no process of this task number which can be performed in the above-mentioned activation table The 1st step which deletes the above-mentioned registration event and registers the head process of the sequence corresponding to this event into the above-mentioned activation table with reference to the above-mentioned sequence information as a process of the above-mentioned task number which can be performed, When the process of the same task number as a current index which can be performed is registered into the above-mentioned activation table The 2nd step deleted from the above-mentioned activation table while performing this process, When there is degree treatment process with reference to the above-mentioned sequence information based on the activation result of the above-mentioned process It has the 3rd step which registers this following treatment process into the above-mentioned activation table as a process which can be performed, and the 4th step which shifts an index, and the program which repeats the above 1st thru/or the 4th step is recorded.

[0071] Moreover, the sequence control program documentation medium concerning claim 50 It has data with which two or more treatment processes contained in a sequence and the transition place from this treatment process were described. The program which is the sequence control program documentation medium which recorded the program which performs the above-mentioned sequence control, reads the contents of the above-mentioned data, changes into a program, and is stored in the storage section, The program which carries out sequential execution of each process of sequence control with reference to the program of the above-mentioned storage section, and the program which performs degree treatment process with reference to the above-mentioned program based on the activation result of the above-mentioned process are recorded.

[0072] Moreover, the record medium concerning claim 51 is a record medium which recorded as data two or more treatment processes contained in a sequence, and the transition place from this treatment process, and is equipped with the 1st field which comes to record two or more above-mentioned treatment processes, and the 2nd field which relates with each of two or more above-mentioned treatment processes, and comes to record an unit or two or more transition place processes.

[0073] Moreover, the record medium concerning claim 52 is equipped with the 3rd field where the above-mentioned record medium comes to record the latency time of the above-mentioned process in relation to the above-mentioned process further in a record medium according to claim 51.

[0074] Moreover, the event corresponding to a task name and this task name in the record medium concerning claim 53, The sequence corresponding to this event, and two or more treatment processes which can be set to this sequence, The 1st field which is the record medium which recorded the transition place from this treatment process as data, and comes to record the above-mentioned task name, The 2nd field which relates with the above-mentioned task name and comes to record an unit or two or more events, It relates with each of the above-mentioned event, and has the 3rd field which comes to record a sequence. The 3rd field of the above Furthermore, the process record section which comes to record the process which relates with each of the above-mentioned sequence and is included in the above-mentioned sequence, the transition place process record

section which the above-mentioned process is alike, respectively, and relates and comes to record an unit or two or more transition place processes -- since -- it is constituted.
[0075]
[Embodiment of the Invention] (Gestalt 1 of operation)
[0076] Hereafter, the task-scheduling equipment by the gestalt 1 of operation of this invention and an approach are explained, referring to a drawing. Drawing 1 is the block diagram showing the configuration of the task-scheduling equipment of the gestalt 1 of this operation. Drawing 2 R> 2 is drawing showing a task activation table. Drawing 3 is a flow chart which shows the task-scheduling approach of the gestalt 1 this operation. The task-scheduling equipment 100 shown in drawing 1 is equipped with the task registration section 101 and the task-scheduling section 102. Furthermore, the task-scheduling section 102 is equipped with the task activation table 103 and the task activation section 104. The task registration section 101 detects the activate request of a task, and registers this task into the task activation table 103 in the task-scheduling section 102. The task activation table 103 is a table which registers temporarily the task which should be performed, for example, is shown by drawing 2 (a) and (b). The above-mentioned task activation table 103 has the magnitude of only the number of tasks, and is making each task and a task number correspond to one to one. The task activation section 104 carries out scheduling of the activation of a task using INDEX and a task number with reference to the task registered into the above-mentioned task activation table 103. Above INDEX is a variable which takes one value of the total task numbers.
[0077] Next, actuation of the task-scheduling equipment 100 of the gestalt 1 of this operation is explained using drawing 2 and drawing 3 . In the gestalt 1 of this operation, CPU with which the task activation section 104 is equipped and which is not illustrated presupposes as an example that processing is performed every 5ms and a maximum of three tasks can be performed by one processing. First, before the processing in CPU which is not illustrated is started, suppose that the task activate request of task A-E occurred. Then, the task registration section 101 detects the activate request of each above-mentioned task, and registers each task into the task of the task activation table 103 which can be performed. The task activation table 103 on which the task was registered has become like drawing 2 (a). In addition, since explanation is easy, suppose that generating of the activate request of each of this task and the registration to the task activation table 103 were happened only once.
[0078] Next, the task-scheduling approach of the task-scheduling section 102 is explained using the flow chart of drawing 3 . CPU which the task activation section 104 does not illustrate -- time amount =0ms -- the 1st processing -- starting -- first -- the number of activation tasks -- it is referred to as =0 and loop count =0 (step S301).
[0079] Next, 1 is added to loop count and it is referred to as loop count =1 (step S302). And whether with reference to the task activation table 103 of drawing 2 (a), the task of the task number corresponding to the present INDEX which can be performed is registered judges the task activation section 104 (step S303), and if registered, it will take out this task, will delete it from the task activation table 103, and will be performed (steps S304 and S305). Here, at the time of processing initiation, only once, INDEX is initialized and it is referred to as INDEX=0. The task whose task number is 0 is Task A, and since Task A is registered into the task which can be performed, Task A is deleted from the task activation table 103, and is performed (step S 304 305). and the number of activation tasks -- 1 -- adding -- the number of activation tasks -- it is referred to as =1 (step S306). In addition, in step S303, when the task of the task number corresponding to current INDEX which can be performed is not registered, step S304 - step S306 are skipped, and it progresses to step S307.
[0080] Next, at step S307, 1 is added to INDEX and it is referred to as INDEX=1. And INDEX judges whether it is more than the number of task activation tables (step S308). if it is more than the number of the task activation tables 103 -- INDEX=0 -- replacing (step S309) -- since it is INDEX=1 this time, replacement of INDEX is not performed.
[0081] Next, it judges whether the number of activation tasks is over the number of the maximum activation (three pieces) (step S310). The number of activation tasks = by 1, the number of the maximum activation is exceeded, since ****, it progresses to step S311 and loop count still judges whether it is under the number of activation tables. Since loop count (1 time) is under the number of task activation tables (= 5), it repeats processing of return and the above flow to step S302. In addition, also when, as for the check of whether this loop count is under the number of activation tables, the task is not registered into the task activation table 103, the task activation section 104 is decision required in order to end processing of that time, when seeing briefly the column of the task of the task activation table 103 which can be performed. If there is no decision of step

S311, when there will be no registration of a task in the task activation table 103, the task activation section 104 checks the task activation table 103 eternally, and cannot end processing of the time.

[0082] thus -- if Task A, Task B, and Task C are performed -- the number of activation tasks -- it is set to =3 and ended by decision of step S310. At this time, Task A, Task B, and Task C are deleted from the task activation table 103.

[0083] It is INDEX=3 when the 1st processing is completed. Moreover, the task activation table 103 when the 2nd processing is started (time amount = 5ms) has become like drawing 2 (b). In addition, in drawing 2 (b), it is shown that the place whose column of the task which can be performed is "-" does not have the task registered. In the 2nd processing by CPU which the task activation section 104 does not illustrate, it performs from Task D, Task D and Task E are performed, and the processing which is the 2nd time is ended. The task activation section 104 merely performs only the check of the task which can be performed every 5ms until there is registration of the new task to the task activation table 103, since the task activation table 103 is the state of the sky after that.

[0084] Thus, without according to the task-scheduling approach by the gestalt 1 of this operation, constituting a complicated program and performing fine control by having considered as the approach of performing the task of the task number corresponding to this INDEX, giving a task number to a task and shifting INDEX in order, it is possible to carry out scheduling of the task, and it can also prevent that a specific task remains not performing. Moreover, since the configuration of a program becomes easy, the size of a program also becomes small. Furthermore, since the number of the maximum tasks performed by one processing is controllable by step S310 of drawing 3 R> 3 according to the capacity of CPU, activation of a task cannot be completed within 1 time of the processing time, but the situation that the starting timing of a resident task and a fixed-cycle task is not securable can also be avoided.

[0085] In addition, although the number of task activation tables is one, and all tasks shall be registered into the above-mentioned task activation table by the task-scheduling equipment of the gestalt 1 of this operation, and the approach when the activate request of a task occurs It is also possible to set up the number of activation tasks which this is an example, for example, divides all tasks into the group of two or more tasks, and is equipped with a task activation table for this every group, and is performed by one task executive operation for every group. At this time, from initiation of the flow chart of drawing 3 to termination will be repeated for every group of each task in one task executive operation. Therefore, when the number of the groups of the above-mentioned task is three, it is necessary to prepare three kinds of INDEX(s) like INDEX (1), INDEX (2), and INDEX (3) for every group. In addition to the same effectiveness as the gestalt 1 of this operation, a task with the above-mentioned high priority has the effectiveness which comes to be performed more certainly by assigning more numbers of activation tasks to the group of a task with a still higher priority by having a task activation table for every specific group of a task as mentioned above, and setting up the number of activation tasks for every above-mentioned group. Moreover, it is also possible to carry out the group division of the task by drive classification of a task, and there is effectiveness performed by being stabilized at this time, without a fixed-cycle task and a resident task being barred by the activation of an event task started irregularly.

[0086] Moreover, although [the gestalt 1 of this operation / INDEX] a task number and INDEX are continuous integers which begin from 0, and INDEX is shifted when only 1 makes this INDEX increase The figure which changes with some fixed regulations, such as a continuous integer which this is an example, for example, begins from 1, or continuous even number which begins from 0, and not overlapping, Or if it is a notation etc., it will be possible to also use them as a task number and INDEX and for it to be possible, and to decrease INDEX further, or to also make it change according to a certain fixed regulation, and the same effectiveness will be acquired.

[0087] Moreover, although Ushiro of activation of a task is performing processing which shifts one INDEX with the gestalt 1 of this operation, it is also possible to perform the processing which this is an example, for example, shifts one INDEX in drawing 3 , step S308, and step S309 in front of the activation S303 of a task, i.e., a step, etc., and the same effectiveness is acquired.

[0088] Moreover, with the gestalt 1 of this operation, although processing spacing of task activation was set to 5ms, this is an example, no matter it may be what processing spacing, scheduling of a task can be realized and equivalent effectiveness is acquired.

[0089] Moreover, although [the gestalt 1 of this operation] task executive operation is started periodically, even if such, the scheduling of the task using a task number and INDEX is possible [this is an example, for example, it is also possible to perform task executive operation continuously, and].

[0090] Moreover, although [the gestalt 1 of this operation] a task is taken out from a task activation table, is deleted and is performed in the flow chart of drawing 3 , this is an example, and after it performs a task, it may delete this task from the above-mentioned task activation table.
[0091] (Gestalt 2 of operation)
[0092] Hereafter, the sequence control equipment by the gestalt 2 of operation of this invention and an approach are explained, referring to a drawing. Drawing 4 is the block diagram showing the configuration of the sequence control equipment 400 by the gestalt 2 of this operation. Drawing 5 is drawing showing the structure of data, and drawing 6 is drawing showing a sequence table and a transition table. Drawing 7 is drawing showing an example of an activation table, a sequence table, and a transition table. The sequence control equipment 400 shown in drawing 4 is equipped with data division 401, the program generation section 402, the storage section 403, and the program execution section 404. Moreover, data division 401 are equipped with a sequence table 405 and the transition table 406. The storage section 403 is equipped with the source code storing section 412, the executive program storing section 413, and memory 414. The program generation section 402 is equipped with the interpretation section 407 and the generation section 408. The program execution section 404 is equipped with the process registration section 409, the activation table 410, and the processes run section 411.
[0093] It is data which have a layered structure as indicated to be a sequence table 405 and the transition table 406 to drawing 5 . When expressed as a table, the transition table 406 is shown like drawing 6 (b) like drawing 6 (a) in a sequence table 405. In the sequence table of drawing 6 (a), a process is processing which CPU performs and the latency time is time amount for which it waits until it performs the above-mentioned process. Moreover, the transition table of drawing 6 (b) is a table showing the transition place process which changes when the activation result of the above-mentioned process is not normal.
[0094] The interpretation section 407 interprets to what kind of process the data indicated there correspond with reference to the data indicated by data division 401. The generation section 408 describes the contents of the data interpreted in the above-mentioned interpretation section 407 with programming language, and stores the generated program source code in the source code storing section 412 of the storage section 403. The program source code stored in the source code storing section 412 of the storage section 403 is compiled and linked, and serves as an executable program, and this executable program is stored in the executive program storing section 413. The above-mentioned executable program is stored in memory 414 at the time of activation. The process registration section 409 registers a process into the activation table 410 with reference to the above-mentioned sequence table 405 stored in the above-mentioned memory 414, and the above-mentioned transition table 406. The activation table 410 is equipped with the cel which can write in one process which can be performed, and the cel which can write in the latency time corresponding to this process as shown by drawing 7 (a). The processes run section 411 performs the process registered into the process which can be performed with reference to the above-mentioned activation table 410. Moreover, reception and the process registration section 409 are made to register a new process into the activation table 410 for the activation result of the above-mentioned process.
[0095] Next, actuation of the sequence control equipment by the gestalt 2 of this operation and the sequence control approach are explained using drawing 8 . Drawing 8 is a flow chart which shows the sequence control approach. Hereafter, the same sequence control as the program generation method of the conventional example is explained as an example. First, the process of the same sequence control as the above-mentioned conventional example etc. is written down in the sequence table 405 and the transition table 406 of data division 401. For example, using spreadsheet software etc., as shown in drawing 7 (b), (c), (d), and (e), each above-mentioned table is created by entering a concrete process name, the latency time, a transition place, etc. in each cel.
[0096] Next, the interpretation section 407 of the program generation section 402 interprets what kind of processes they are concretely with reference to a process name, a transition place name, etc. which were indicated by each table of the above-mentioned data division 401. for example, a spreadsheet -- a character string like drawing 7 (b) indicated in the soft table etc. is read, and it interprets of what kind of contents the "processing A" etc. is processing concretely. And the interpretation result is told to the generation section 408. The generation section 408 describes the table data of the layered structure indicated by data division 401 with programming language based on the interpretation result from the above-mentioned interpretation section 407, and stores the generated program source code in the source code storing section 412 of the storage section 403. Therefore, the program itself currently recorded on the storage section 403 is only what described the data of the layered structure shown in drawing 5 with programming language. The above-mentioned program source

code is compiled and linked, serves as an executable program, and is stored in the executive program storing section 413. This executable program is stored in memory 414 at the time of activation. The program execution section 404 performs sequence control, referring to the program stored in the above-mentioned memory 414.
[0097] Hereafter, the above-mentioned sequence control is explained using the flow chart of drawing 8 . First, an instruction is issued so that the processes run section 411 may register a head process and its latency time into the process of the activation table 410 which can be performed at the process registration section 409. Then, the process registration section 409 reads the head process and the latency time of a sequence table 405 with reference to memory 414, and registers them into the activation table 410 (step S801). The activation table 410 on which the processing A which is the head process of the sequence table of drawing 7 (b) was registered is shown by drawing 7 (a).
[0098] Next, the latency time of the process in which the processes run section 411 was registered into the process of the activation table 410 which can be performed = it judges in 0 (S802). Since the latency time of Processing A is 0, Processing A is taken out and deleted from the activation table 410 (step S804).
[0099] Next, it progresses to step S805 and processing A taken out from the activation table 410 is performed. The activation result of Processing A is returned to the processes run section 411. As for the processes run section 411, this activation result judges whether it is normal (step S806). Although the column of a transition table is "-" about Processing A, it is shown that this does not have a setup of a transition table and all activation results become normal. Therefore, when whether a process [degree] process is in the process registration section 409 makes the processes run section 411 judge and there is a process [degree] process, it makes this process register into the activation table 410. the process registration section 409 has a process [degree] process "conditional judgment 1" in a sequence table with reference to memory 414 -- checking (step S807) -- this -- a process [degree] process and its latency time are registered into the activation table 410 (step S808).
[0100] And the latency time corresponding to the conditional judgment 1 by which return and the processes run section 411 were registered into step S802 by the process of the activation table 410 which can be performed = it judges whether it is 0. Since the latency time of conditional judgment 1 is 20ms, waiting for time amount is carried out for 20ms (step S803), and conditional judgment 1 is taken out from the activation table 410, is deleted, and is performed (steps S804 and S805).
[0101] The activation result of the above-mentioned process is returned to the processes run section 411, and this activation result judges whether it is normal (step S806). In activation of the above-mentioned conditional judgment 1, when an activation result becomes normal when fulfilling conditions 1, and you do not fulfill conditions 1, suppose that an activation result is not normal. When an activation result is not normal, the processes run section 411 issues an instruction so that a transition place process may be registered into the activation table 410 at the process registration section 409 with reference to the transition table corresponding to the above-mentioned conditional judgment 1. The process registration section 409 registers into the activation table 410 the conditional judgment 2 whose process number is 3 with reference to the transition table A of drawing 7 (c) of the Records Department 403. And the processing from step S802 is repeated again.
[0102] In the above-mentioned step S806, when it judges that the activation result of the above-mentioned conditional judgment 1 has the normal processes run section 411, an instruction is issued so that a process [degree] process may be registered into the activation table 410 at the process registration section 409, and the above-mentioned process registration section 409 registers into the activation table 410 the processing 1 which is a process [degree] process. And the processing from step S802 is repeated again. Thus, if the processes run section 411 performs even processing C whose process number is 7, since there is no process [degree] process, it will be that it is ended.
[0103] Thus, according to the sequence control equipment in the gestalt 2 of this operation, and the approach, the program itself is only data of a layered structure, it is having had the program execution section 404 which can perform the program of only the data, and can perform the C tense control of the program of only the above-mentioned data. Moreover, as mentioned above, since it can respond to the addition of the process to this program, deletion, modification, etc. easily since a program is only data, and a change of the transition place as conditional branching can also be easily made only by modification of the contents of the transition table, the adaptation range is wide and the high sequence control equipment of description capacity and an approach can be offered. For example, what is necessary is just to add Processing D to the sequence table of drawing 7 (b) as a sequence number 8 in the example of drawing 7 to add Processing D after Processing C.
[0104] Furthermore, a program will be only data, and since conditional branching etc. is not included in a program, even if program capacity serves as necessary minimum, and the capacity of a program decreases

compared with the conventional thing and it adds processing, the increment in the capacity of a program will require only a part for the data of the processing. Therefore, versatility is very high and serves as sequence control equipment which can respond to various sequence control flexibly, and an approach. as mentioned above, compared with the conventional thing, the size effectiveness of a program is boiled markedly and is improving.

[0105] In addition, although the transition place process specified on a transition table was set to one in the program generation method of the gestalt 2 of this operation, this is an example, in decision of step S806 of the flow chart of drawing 8 , when the activation result of a process is not normal, it is also possible to choose either from two or more transition places, and the same effectiveness is acquired. For example, in the sequence table of drawing 7 (b), change the transition table A into transition table A' of drawing 7 (f), and it sets to activation of the conditional judgment 1 of a sequence number 1. Although a result presupposes that it is normal and is not applied to conditions 1, when applied to conditions 1, when applied to conditions 2, Although it is not applied to conditions 1 and conditions 2, when applied [to conditions 3] and not applied to conditions 1, conditions 2, and conditions 3, it can also set up, respectively so that it may change in the process of the transition number 1, the transition number 2, and the transition number 3. Thus, a transition place can be easily changed only by modification of a transition table.

[0106] Moreover, although [the gestalt 2 of this operation / data division 401 the program generation section 402, the storage section 403, and the program execution section 404] all are contained in one sequence control equipment 400 This is an example and the program recorded on the program generation equipment 900 which generates a program, the record medium 903 which records the program by which generation was carried out [above-mentioned], and the above-mentioned record medium 903 is read like drawing 9 R> 9. It is also possible to divide into the program execution equipment 901 which performs sequence control. In addition, let the approach of performing sequence control for the approach of generating a program with the above-mentioned program generation equipment 900 with a program generation method and the above-mentioned program execution equipment 901 be the program execution approach. Here, in drawing 9 , the same sign as drawing 4 shows the same thing as the sequence control equipment of the gestalt 2 of this operation, and those explanation is omitted. The write-in section 902 writes the program which the program generation section 402 generated in a record medium 903. An executable program is written in the above-mentioned record medium 903. Moreover, the reading section 904 reads the executable program written in the record medium 903. The read program is stored in the memory which the above-mentioned reading section 904 does not illustrate.

[0107] Since only program execution equipment 901 can be further used independently on the same effectiveness as the gestalt 2 of this operation by dividing the sequence control equipment 400 of the gestalt 2 of this operation into program generation equipment 900 like drawing 9 , program execution equipment 901, and a record medium 903, the part and the equipment which do not contain program generation equipment 900 compared with sequence control equipment 400 can be miniaturized. Moreover, since it is also possible to realize sequence control with two or more program execution equipments 901 even if there is only single program generation equipment 900, the cost can be cut down compared with the case where two or more sequence control equipments 400 are used. Furthermore, the sequence control of program execution equipment 901 can also be easily changed only by modification of a record medium 903 by recording the data program of the sequence control of two or more classes on the record medium 903 with program generation equipment 900 in advance.

[0108] Moreover, with the gestalt 2 of this operation, the process registration section 409 accesses the executable program stored in the memory 414 of the storage section 403. Although the process included in this executable program shall be registered into the activation table 410 and the processes run section 411 shall perform the process by which registration was carried out [above-mentioned] This is an example, it is also possible to perform sequence control, not having a component which registers a process temporarily as shown in the above-mentioned activation table 410, but accessing the memory 414 of the storage section 403 directly, and the same effectiveness is acquired.

[0109] Moreover, with the gestalt 2 of this operation, although the number of sequence tables was set to one, this is an example and preparing two or more sequence tables, or also considering as the sequence of a layered structure by [of a sequence table] preparing the subsequence table of the hierarchy under one more etc. further and the effectiveness that it is possible and the description capacity other than the same effectiveness as the gestalt 2 of this operation becomes high further are acquired.

[0110] Moreover, with the gestalt 2 of this operation, although the latency time corresponding to a process was

set up, this is an example, for example, it is also possible to perform time amount waiting of only the necessary latency time as one process occasionally whose latency time is the need, without setting up the latency time, and the same effectiveness is acquired.

[0111] Moreover, although [the gestalt 2 of this operation] a process is taken out from an activation table, is deleted and is performed in the flow chart of drawing 8 , this is an example, and after it performs a process, it may delete this process from the above-mentioned activation table. Or overwriting is also possible, in case the process to delete is skipped and a process [degree] process or a transition place process is registered into an activation table.

[0112] Moreover, with the gestalt 2 of this operation, a process [degree] process is an unit, although the transition place process should be chosen as the transition table based on the activation result from the multiple processes of a publication, this is an example, it is also possible to choose a process [degree] process when the activation result of a process is normal from multiple processes, and the same effectiveness is acquired.

[0113] (Gestalt 3 of operation)

[0114] The sequence control equipment by the gestalt 3 of operation of this invention and an approach perform sequence control included in the above-mentioned task, carrying out scheduling of two or more tasks by INDEX. Hereafter, the sequence control equipment of the gestalt 3 of this operation and an approach are explained, referring to a drawing. Drawing 10 is the block diagram showing the configuration of the sequence control equipment of the gestalt 3 of this operation. Sequence control equipment 1000 is equipped with data division 1001, the program generation section 1002, the storage section 1003, and the program execution section 1004. Moreover, data division 1001 are equipped with the task table 1005, the event table 1006, a sequence table 1007, the subsequence table 1008, and the transition table 1009. The program generation section 1002 is equipped with the interpretation section 1010 and the generation section 1011. The storage section 1003 is equipped with the source code storing section 1017, the executive program storing section 1018, and memory 1019. The program execution section 1004 is equipped with the event analysis section 1012, the event queue table 1013, the sequence activation section 1014, the process registration section 1015, and the activation table 1016.

[0115] Drawing 11 is drawing having shown the data-hierarchy structure of the task table 1005, the event table 1006, a sequence table 1007, and the subsequence transition table 1008 and 1009. Drawing 12 is drawing having shown each above-mentioned table. The task table 1005 of drawing 12 (a) indicates correspondence with the event table 1006 to be each task. The event table 1006 of drawing 12 (b) shows correspondence with the event belonging to this event table 1006, and event reception conditions and a sequence table 1007.

[0116] When the above-mentioned event reception conditions are "with no queue", in case the generated event is registered into the event queue corresponding to the above-mentioned event of the event queue table 1013, it registers, after deleting the event by which the queuing was carried out by then, and the generated event is registered into the last edge of the above-mentioned event queue when the above-mentioned event reception conditions are "with a queue." Therefore, the urgent event reception conditions of an event presuppose "with no queue", and let the event reception conditions of the low event of an urgency be "those with a queue."

[0117] The sequence table 1007 of drawing 12 (c) has further two or more subsequence tables 1008. The subsequence table 1008 of drawing 12 (d) has multiple processes further, and the latency time, transition authorization, and the transition table 1009 support this process. It is the above-mentioned process which CPU actually performs in sequence control. The above-mentioned latency time is the latency time until activation of the process is started.

[0118] Moreover, when the above-mentioned transition authorization is "authorization" and an event is in an event queue, the sequence under activation is ended. Register the process corresponding to the above-mentioned event into the activation table 1016, perform it, and when the above-mentioned transition authorization is "prohibition" or [that the process of "authorization" is registered into the activation table 1016 for transition authorization in the sequence under activation even when an event is in an event queue] -- or The sequence corresponding to the above-mentioned event cannot be started until all the processes of the above-mentioned sequence are completed.

[0119] The transition table 1009 of drawing 12 (e) shows the transition place process which changes when the activation result of the above-mentioned process is not normal by the sequence table 1007, the subsequence number, and the process number. The interpretation section 1010 interprets whether with reference to the data indicated by data division 1001, the contents indicated there are equivalent to what kind of process. The generation section 1011 describes the contents of the data interpreted in the above-mentioned interpretation

section 1010 with programming language, and stores the generated program source code in the source code storing section 1017 of the storage section 1003. The program source code stored in the source code storing section 1017 of the storage section 1003 is compiled and linked, and serves as an executable program, and this executable program is stored in the executive program storing section 1018. The above-mentioned executable program is stored in memory 1019 at the time of activation. The event analysis section 1012 detects that the event occurred, and carries out the queuing of the event concerned to the event queue corresponding to the above-mentioned event of the event queue table 1013.

[0120] Each event queue of the event queue table 1013 stores the generated event. Drawing 13 (a) is drawing having shown this event queue table 1013. Each task has an event queue corresponding to this task. That is, an event queue has only the class of task. An event queue is the DS of FIFO (First In First Out), the generated event is registered behind the registered event of the event queue corresponding to this event, and the event registered is taken out and deleted from the head side. If a top event is deleted, the event registered into the degree will shift to a head side by one.

[0121] The activation table 1016 has the magnitude of only the number of tasks, and the process which each task and a task number should be made to correspond to one to one, and then should perform them and the latency time corresponding to it, and transition authorization are stored temporarily. Drawing 13 (b) is drawing showing the above-mentioned activation table 1016. With reference to the above-mentioned event queue activation table 1013 and 1016, the sequence activation section 1014 carries out scheduling of two or more tasks using the above-mentioned task number and INDEX, and performs a process. Above INDEX is a variable which takes one value of the total task numbers. The process registration section 1015 registers a process into the activation table 1016, referring to the data program of memory 1019 based on the instruction from the above-mentioned sequence activation section 1014.

[0122] Next, actuation of the sequence control equipment 1000 of the gestalt 3 of this operation and the sequence control approach are explained using drawing 16 , drawing 17 , and drawing 18 . Drawing 16 is the flow chart which showed actuation of the event analysis section 1012. Drawing 17 is the flow chart which showed actuation of the program execution section 1004. Drawing 18 is the flow chart which showed the predefined process in the flow chart of drawing 17 "a processes run."

[0123] First, the process of the sequence control to wish, the relation between the event table 1006 and a task, etc. are entered in the task table 1005, the event table 1006, the sequence table 1007, and the subsequence transition table 1008 and 1009 of data division 1001 of a layered structure like drawing 11 . For example, each above-mentioned table is created like drawing 12 (a) - (e) using spreadsheet software etc. by entering a concrete process name, the latency time, a table name, etc. in each cel.

[0124] Next, the interpretation section 1010 of the program generation section 1002 interprets the event [what kind of process] they support concretely with reference to a process name, an event name, etc. which were indicated by each table of the above-mentioned data division 1001. And the interpretation result is told to the generation section 1011. The generation section 1011 describes the table data of the layered structure indicated by data division 1001 with programming language based on the interpretation result from the above-mentioned interpretation section 1010, and stores the generated program source code in the source code storing section 1017 of the storage section 1003. And the above-mentioned program source code is compiled and linked, serves as an executable program, and is stored in the executive program storing section 1018. The above-mentioned executable program is stored in memory 1019, and is performed. Therefore, the program itself stored in the storage section 1003 is what [only] described the data of the layered structure shown in drawing 11 by the program source code. That is, it is a data program and the executive program of the sequence control is not contained. Referring to the program recorded on the above-mentioned memory 1019, the program execution section 1004 carries out scheduling of the task using INDEX and a task number, and performs sequence control corresponding to the task.

[0125] Hereafter, actuation of sequence control equipment 1000 is explained by making control of an air-conditioner into an example. Some table data in the air-conditioner control are shown in drawing 14 . First, the table data shown in drawing 14 are changed into a program by the program generation section 1002, and presuppose after that that it was compiled and linked and was stored in memory 1019.

[0126] Next, when a certain event occurs externally, the event analysis section 1012 detects that and carries out the queuing of this event to the event queue corresponding to the above-mentioned event of the event queue table 1013 according to a fixed regulation. Actuation of the event analysis section 1012 is shown by the flow chart of drawing 16 .

[0127] The event analysis section 1012 supervises whether there is any generating of an event continuously as a resident task. And if there is generating of an event, with reference to the event table as a data program currently recorded on memory 1019, it will check first to which event table the generated event belongs. Next, with reference to a task table, correspondence with the above-mentioned event table and a task is checked (step S1601). And when it judges whether there is any registration event to the event queue corresponding to the above-mentioned task of the event queue table 1013 (step S1602) and there is no registration event in it, it progresses to step S1605 and the event generated in the above-mentioned event queue is registered.
[0128] In the example of air-conditioner control, like drawing 15 (a), the air conditioning carbon button of remote control is pushed, and suppose that the event "air conditioning" occurred. As shown in the event queue table 1013 on the right-hand side of drawing 15 (a), supposing processing has still started just and anything does not have an event in the event queue table 1013 at this time, the event analysis section 1012 will carry out the queuing of the event "an air conditioning key" to the head of an air conditioning event queue.
[0129] In step S1602 of drawing 16 , when a registration event is in the event queue corresponding to the generated event, the reception conditions of the generated event judge "he has no queue" and "those with a queue" (step S1603). And when event reception conditions are "with no queue", all the registration events of the above-mentioned event queue are deleted, and the generated event is registered into the head of an event queue. By decision of step S1603, when event reception conditions are "with a queue", the event concerned is registered behind the event progressed and registered into step S1605. As mentioned above, since it is a resident task, the event analysis section 1012 always supervises generating of an event, and when there is generating of an event, it repeats processing from initiation of the flow chart of drawing 16 to termination.
[0130] Next, suppose that the processing in CPU which the sequence activation section 1014 does not illustrate was started. In the gestalt 3 of this operation, executive operation is started every 5ms and suppose that a maximum of three processes can be performed by one processing. if CPU which the sequence activation section 1014 does not illustrate starts processing -- first -- the number of activation processes -- it is referred to as =0 and loop count =0 (step S1701), and next, 1 is added to loop count and it is referred to as loop count =1 (step S1702). And it judges whether the event is registered into the event queue of the task number corresponding to current INDEX of the activation table 1016 (step S1703). Here, at the time of processing initiation of CPU, only once, INDEX is initialized and it is referred to as INDEX=0. Then, since the event is not registered into a corresponding display event queue, it progresses to step S1708 and judges whether registration of the process of the task number corresponding to INDEX which can be performed is shown in the activation table 1016. Since there is also no registration of the process which can be performed, 1 is added to INDEX (step S1710), and it is referred to as INDEX=1. And INDEX judges whether it is more than the number of activation tables (= 3) (step S1711), since it still is not more than the number of activation tables, it progresses to step S1713, and the number of activation processes judges whether it is more than the number of the maximum activation (= 3) (step S1713). Since it still is not more than the number of the maximum activation, it progresses to step S1714 and loop count judges whether it is more than the number of activation tables. Since loop count still is not more than the number of activation tables, either, it adds return even to step S1702, adds 1 to loop count, and sets it to loop count =2. Since there is no event registered into the temperature monitor event queue of the task number corresponding to INDEX=1 this time also, it adds 1 to INDEX (step S1710), and only sets to INDEX=2, and returns to step S1702.
[0131] At the time of INDEX=2, since the queuing of the registration event "an air conditioning key" is carried out, in decision of step S1703, to the corresponding air conditioning event queue of a task number, it is judged as those with a registration event, and judges to it whether there is any task which is registered into the task activation table 1016 next and which can be performed (step S1704). Since nothing is registered into the activation table 1016, it progresses to step S1706 and the sequence activation section 1014 still gives [registering the head process corresponding to the head event of an event queue into the activation table 1016, and] an instruction to the process registration section 1015.
[0132] Since the air conditioning key is registered into the air conditioning event queue, the above-mentioned process registration section 1015 registers the compressor situation check which is the head process of this subsequence table into the process corresponding to the air conditioning task of the activation table 1016 which can be performed with reference to the air conditioning sequence of memory 1019 with reference to the compressor control subsequence table which is the head subsequence of an air conditioning sequence. At this time, the latency time and transition authorization which are indicated by coincidence at this subsequence table are also registered into the activation table 1016. The registered activation tables 1016, such as the above-

mentioned process, are shown by drawing 15 (b). And the sequence activation section 1014 deletes an air conditioning key from an air conditioning event queue (step S1707). Next, it progresses to step S1708, and although it judges whether the process which can be performed is shown in the activation table 1016, since there is a compressor situation check registered previously, those with an activation process, a next door, and predefined process "a processes run" are started (step S1709).

[0133] The predefined process "a processes run" of step S1709 is shown by the flow chart of drawing 18 . First, the latency time of the task number corresponding to the present INDEX of the activation table 1016 in the sequence activation section 1014 = it judges whether it is 0 (step S1801), and when it is not latency-time =0, a processes run is ended, it progresses to step S1710 of the flow chart of drawing 17 R> 7, and 1 is added to INDEX. This time, since it is latency-time =0, a compressor situation check is taken out from the process of the activation table 1016 which can be performed, this process is deleted from the above-mentioned activation table 1016 (step S1802), and this process is performed (step S1803). and the number of activation processes -- 1 -- adding -- the number of activation processes -- it is referred to as =1.

[0134] Next, the activation result of a compressor situation check is returned to the sequence activation section 1014. And it judges whether the sequence activation section 1014 has the normal activation result of the above-mentioned compressor situation check (step S1805), and when not normal, an instruction is given to the process registration section 1015 so that the process of a corresponding transition place may be registered into the activation table 1016 with reference to the transition table corresponding to a compressor situation check. The process registration section 1015 registers a transition place process into the activation table 1016 with this instruction (step S1808).

[0135] Occasionally, it judges [whose activation result is normal] whether the sequence activation section 1014 has a process [degree] process in the process registration section 1015 (step S1806), and in a certain case, an instruction is issued so that the process may be registered into the activation table 1016 (step S1807). Since there is engine-speed modification as a process [degree] process this time, the process registration section 1015 registers this process into the process of the activation table 1016 which can be performed. And the processes run of step S1709 is ended, 1 is added to INDEX, and it is referred to as INDEX=3 (step S1710). And it judges that INDEX is more than the number of activation tables in step S1711, and replaces with INDEX=0.

[0136] Next, it progresses to step S1714 and it is judged that loop count is more than the number of activation tables. In addition, also when, as for decision of step S1714, the process is not registered into the activation table 1016, the sequence activation section 1014 is decision required in order to end processing of the time, when seeing briefly the column of the process of the activation table 1016 which can be performed. If there is no decision of step S1714, when there will be no registration of a process in the activation table 1016, the sequence activation section 1014 checks the activation table 1016 eternally, and cannot end processing of the time. And it progresses to step S1715, only processing spacing time amount (= 5ms) lengthens the latency time of all the processes of the activation table 1016 registered, and this sequence executive operation is ended. Initiation of next sequence executive operation carries out sequential execution of the processing from step S1701.

[0137] Thus, according to the sequence control equipment by the gestalt 3 of this operation, and the approach By carrying out scheduling of the process corresponding to two or more tasks which should be performed with INDEX and a task number, and carrying out sequential execution of the data of a layered structure by the program execution section 1004 The process which remains being able to perform two or more sequence control in parallel, and not performing will not exist, but the size of a data program itself can be further managed with necessary minimum. Moreover, since the data of the gestalt 3 of this operation are only expressed by the table of a layered structure, easily, the addition of a process and deletion are performed and a change etc. can be made, and versatility is high and it becomes possible to realize the sequence control equipment which was rich in flexibility, and an approach. Furthermore, since the number of the maximum processes performed by one processing is controllable by step S1713 of drawing 1717 according to the capacity of CPU, the situation which activation of a process does not end within 1 time of the processing time is also avoidable.

[0138] Moreover, before performing sequence processing corresponding to the event even if other events occur by carrying out transition authorization of the process to prohibition when there is a process which must perform it in a sequence since transition authorization of a subsequence table can be performed with "prohibition" or "authorization", the above-mentioned process can be performed.

[0139] In addition, by the sequence control equipment of the gestalt 3 of this operation, and the approach, although only the sequence control corresponding to an event task was explained, this is an example, for

example, sequence control corresponding to a resident task and a fixed-cycle task can also be performed, and the same effectiveness is acquired. As the approach of the sequence control of a resident task and a fixed-cycle task, it has the periodic event generator made to generate an event periodically, or the process [degree] process of the process of the last of a sequence is made into the head process of a sequence, and, in the case of a resident task, the latency time of this head process is set to 0, and, in the case of a fixed-cycle task, the latency time of this head process should just be determined in consideration of a period.

[0140] Moreover, although the number of activation tables shall be one and the process corresponding to all tasks shall be registered into the above-mentioned activation table with the gestalt 3 of this operation in registration of the process which can be performed It is also possible to set up the number of activation processes which this is an example, for example, divides all tasks into the group of two or more tasks, and is equipped with an activation table for this every group, and is performed by one processing for every group. At this time, from initiation of the flow chart of drawing 17 R> 7 to termination will be repeated for every group of a task in one processing. Therefore, when the number of the groups of the above-mentioned task is three, it is necessary to prepare three kinds of INDEX(s) like INDEX (1), INDEX (2), and INDEX (3) for every group. In addition to the same effectiveness as the gestalt 3 of this operation, a sequence with the above-mentioned high priority has the effectiveness performed more certainly by assigning more numbers of activation processes to the group of a task including a sequence with a still higher priority by having an activation table for every specific group of a task as mentioned above, and setting up the number of activation processes for every above-mentioned group. Moreover, it is also possible to carry out the group division of the task by drive classification of a task, and there is effectiveness performed by being stabilized at this time, without a fixed-cycle task and a resident task being barred by the activation of an event task started irregularly.

[0141] Moreover, with the gestalt 3 of this operation, although sequence control was expressed with two hierarchies of a sequence table 1007 and the subsequence table 1008, this is an example, and it is only a sequence table according to the amount of the process included in sequence control, or it is possible to realize also by using three or more hierarchies' sequence table, a subsequence table, etc., and the same effectiveness is acquired.

[0142] Moreover, although [the gestalt 3 of this operation / data division 1001 the program generation section 1002, the storage section 1003, and the program execution section 1004] all are contained in one sequence control equipment 1000 This is an example and the program recorded on the program generation equipment 1900 which generates a program, the record medium 1903 which records the program by which generation was carried out [above-mentioned], and the above-mentioned record medium 1903 is read like drawing 19 . It is also possible to divide into the program execution equipment 1901 which performs sequence control. In addition, let the approach of performing sequence control for the approach of generating a program with the above-mentioned program generation equipment 1900 with a program generation method and the above-mentioned program execution equipment 1901 be the program execution approach. Here, in drawing 19 , the same sign as drawing 10 shows the same thing as the sequence control equipment of the gestalt 3 of this operation, and those explanation is omitted. The write-in section 1902 writes the program which the program generation section 1002 generated in a record medium 1903. An executable program is written in the above-mentioned record medium 1903. Moreover, the reading section 1904 reads the executable program written in the record medium 1903. The read program is stored in the memory which the above-mentioned reading section 1904 does not illustrate.

[0143] Since only program execution equipment 1901 can be further used independently on the same effectiveness as the gestalt 3 of this operation by dividing the sequence control equipment 1000 of the gestalt 3 of this operation into program generation equipment 1900 like drawing 19 , program execution equipment 1901, and a record medium 1903, the part and the equipment which do not contain program generation equipment 1900 compared with sequence control equipment 1000 can be miniaturized. Moreover, since it is also possible to realize sequence control with two or more program execution equipments 1901 even if there is only single program generation equipment 1900, the cost can be cut down compared with the case where two or more sequence control equipments 1000 are used. Furthermore, the sequence control of program execution equipment 1901 can also be easily changed only by modification of a record medium 1903 by recording the data program of the sequence control of two or more classes on the record medium 1903 with program generation equipment 1900 in advance.

[0144] Moreover, although the number of the events which can carry out a queuing to each event queue of the event queue table 1013 is written by drawing 13 (a) as three pieces with the gestalt 3 of this operation, if it is the

number which can carry out the queuing only of the required event by the system, this will be an example, it is not restricted to three pieces, and you may be four pieces and five pieces or more, for example, the same effectiveness will be acquired.

[0145] Moreover, with the gestalt 3 of this operation, although the transition place process on the transition table 1009 shall be specified by assignment of a transition place sequence table, a subsequence number, and a process number, this is an example, for example, by specifying a subsequence table and a process, a transition place process can also be specified and the same effectiveness is acquired.

[0146] Moreover, although [the gestalt 3 of this operation / INDEX] a task number and INDEX are continuous integers which begin from 0, and INDEX is shifted when only 1 makes this INDEX increase The figure which changes with some fixed regulations, such as a continuous integer which this is an example, for example, begins from 1, or continuous even number which begins from 0, and not overlapping, Or if it is a notation etc., it will be possible to also use them as a task number and INDEX and for it to be possible, and to decrease INDEX further, or to also make it change according to a certain fixed regulation, and the same effectiveness will be acquired.

[0147] Moreover, although Ushiro of activation of a process is performing processing which shifts one INDEX with the gestalt 3 of this operation, it is also possible to perform the processing which this is an example, for example, shifts one INDEX in drawing 17 , step S1711, and step S1712 in front of the activation S1708 of a process, i.e., a step, etc., and the same effectiveness is acquired.

[0148] Moreover, with the gestalt 3 of this operation, although processing spacing of a processes run was set to 5ms, this is an example, no matter it may be what processing spacing, control of the sequence included in two or more tasks can be realized, and equivalent effectiveness is acquired.

[0149] Moreover, although [the gestalt 3 of this operation] processes run processing is performed periodically, even if such, the scheduling of the task using a task number and INDEX is possible [this is an example, for example, it is also possible to perform processes run processing continuously, and].

[0150] Moreover, although [the gestalt 3 of this operation] a process is taken out from an activation table, is deleted and is performed in the flow chart of drawing 18 , this is an example, and after it performs a process, it may delete this process from the above-mentioned activation table. Or overwriting is also possible, in case the process to delete is skipped and a process [degree] process or a transition place process is registered into an activation table.

[0151] Moreover, with the gestalt 3 of this operation, a process [degree] process is an unit, although the transition place process should be chosen as the transition table based on the activation result from the multiple processes of a publication, this is an example, it is also possible to choose a process [degree] process when the activation result of a process is normal from multiple processes, and the same effectiveness is acquired.

[0152] In addition, also when the record medium which recorded the program which realizes the task-scheduling approach shown with the gestalt of each above-mentioned implementation, the sequence-control approach, a program generation method, and the program execution approach supplies to a system or equipment and the main processing sections, such as the CPU of the system or equipment, read and perform the program stored in this record medium, the effectiveness which explained with the gestalt of each above-mentioned implementation, and the same effectiveness can acquire.

[0153] In addition, as a record medium which records a program, a floppy disk, a hard disk, an optical disk, a magnetic disk, a magneto-optic disk, CD-ROM, a magnetic tape, a punch card, the memory card of a non-volatile, ROM, etc. can be used, for example.

[Translation done.]

* NOTICES *

1.This document has been translated by computer. So the translation may not reflect the original precisely.
2.**** shows the word which can not be translated.
3.In the drawings, any words are not translated.

DESCRIPTION OF DRAWINGS

[Brief Description of the Drawings]
[Drawing 1] It is the block diagram showing the configuration of the task-scheduling equipment by the gestalt 1 of operation of this invention.
[Drawing 2] It is drawing showing the task activation table by the gestalt of this operation ((a), (b)).
[Drawing 3] It is the flow chart which shows the task-scheduling approach by the gestalt of this operation.
[Drawing 4] It is the block diagram showing the configuration of the sequence control equipment by the gestalt 2 of operation of this invention.
[Drawing 5] It is drawing showing the structure of the data based on the gestalt of this operation.
[Drawing 6] It is drawing showing the (a) sequence table by the gestalt and (b) transition table of this operation.

[Drawing 7] It is drawing showing (a) activation table by the gestalt of this operation, the (b) sequence table, and (c) - (d) transition table.
[Drawing 8] It is the flow chart by the gestalt of this operation which shows the sequence control approach.
[Drawing 9] It is the block diagram showing the configuration of the program generation equipment by the gestalt of this operation, a record medium, and program execution equipment.
[Drawing 10] It is the block diagram showing the configuration of the sequence control equipment by the gestalt 3 of operation of this invention.
[Drawing 11] It is drawing by the gestalt of this operation showing the structure of data.
[Drawing 12] It is drawing showing (a) task table by the gestalt, (b) event table, the (c) sequence table, (d) subsequence table, and (e) transition table of this operation.
[Drawing 13] It is drawing showing the (a) event queue by the gestalt and (b) activation table of this operation.
[Drawing 14] It is drawing showing (a) task table by the gestalt, (b) event table, the (c) sequence table, (d) subsequence table, and (e) transition table of this operation.
[Drawing 15] It is drawing showing (a) remote control by the gestalt, the event analysis section, event queue, and (b) activation table of this operation.
[Drawing 16] It is the flow chart by the gestalt of this operation which showed actuation of the event analysis section.
[Drawing 17] It is the flow chart by the gestalt of this operation which showed the sequence control approach.
[Drawing 18] It is the flow chart by the gestalt of this operation which showed the predefined process in the sequence control approach, and the processes run section.
[Drawing 19] It is the block diagram showing the configuration of the program generation equipment by the gestalt of this operation, a record medium, and program execution equipment.
[Drawing 20] It is drawing having shown the execution priority of the (a) task by the conventional task-scheduling approach, and (b) (c) task activation table.
[Drawing 21] It is drawing having shown (a) sequence sequence table by the conventional task-scheduling approach, and the operating-condition table of the (b) processing B.
[Drawing 22] It is the flow chart which showed the program generation method using the conventional fixed sequence pattern.
[Drawing 23] It is drawing by the program generation method using the conventional state transition table showing a state transition table.
[Drawing 24] It is the flow chart which showed the program generation method using the conventional state transition table.

[Description of Notations]
100 Task-Scheduling Equipment
101 Task Registration Section
103 Task Activation Table
104 Task Activation Section
400 Sequence Control Equipment
401 Data Division
402 Program Generation Section
403 Storage Section
404 Program Execution Section
405 Sequence Table
406 Transition Table
407 Interpretation Section
408 Generation Section
409 Process Registration Section
410 Activation Table
411 Processes Run Section
412 Source Code Storing Section
413 Executive Program Storing Section
414 Memory
900 Program Generation Equipment
901 Program Execution Equipment
902 Write-in Section
903 Record Medium
904 Reading Section
1000 Sequence Control Equipment
1001 Data Division
1002 Program Generation Section
1003 Storage Section
1004 Program Execution Section
1005 Task Table
1006 Event Table
1007 Sequence Table
1008 Subsequence Table
1009 Transition Table
1010 Interpretation Section
1011 Generation Section
1012 Event Analysis Section
1013 Event Queue Table
1014 Sequence Activation Section
1015 Process Registration Section
1016 Activation Table
1017 Source Code Storing Section
1018 Executive Program Storing Section
1019 Memory
1900 Program Generation Equipment
1901 Program Execution Equipment
1902 Write-in Section
1903 Record Medium
1904 Reading Section

[Translation done.]